

the algorithm is fairly general and performs remarkably well with most types of images.

ACKNOWLEDGMENT

The author would like to thank Joel Zdepksi who suggested incorporating the sign of the significant values into the significance map to aid embedding, Rajesh Hingorani who wrote much of the original C code for the QMF-pyramids, Allen Gersho who provided the original "Barbara" image, and Gregory Wornell whose fruitful discussions convinced me to develop a more mathematical analysis of zerotrees in terms of bounding an optimal estimator. I would also like to thank the editor and the anonymous reviewers whose comments led to a greatly improved manuscript.

REFERENCES

- [1] E. H. Adelson, E. Simoncelli, and R. Hingorani, "Orthogonal pyramid transforms for image coding," *Proc. SPIE*, vol. 845, Cambridge, MA, Oct. 1987, pp. 50-58.
- [2] R. Ansari, H. Gaggioni, and D. J. LeGall, "HDTV coding using a nonrectangular subband decomposition," in *Proc. SPIE Conf. Visual Commun. Image Processing*, Cambridge, MA, Nov. 1988, pp. 821-824.
- [3] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1990.
- [4] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, pp. 532-540, 1983.
- [5] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Informat. Theory*, vol. 38, pp. 713-718, Mar. 1992.
- [6] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, pp. 909-996, 1988.
- [7] —, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. Informat. Theory*, vol. 36, pp. 961-1005, Sept. 1990.
- [8] R. A. DeVore, B. Jawerth, and B. J. Lucier, "Image compression through wavelet transform coding," *IEEE Trans. Informat. Theory*, vol. 38, pp. 719-746, Mar. 1992.
- [9] W. Equitz and T. Cover, "Successive refinement of information," *IEEE Trans. Informat. Theory*, vol. 37, pp. 269-275, Mar. 1991.
- [10] Y. Huang, H. M. Driksen, and N. P. Galatsanos, "Prioritized DCT for Compression and Progressive Transmission of Images," *IEEE Trans. Image Processing*, vol. 1, pp. 477-487, Oct. 1992.
- [11] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [12] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing*, vol. 1, pp. 31-48, Jan. 1992.
- [13] J. Kovačević and M. Vetterli, "Nonseparable multidimensional perfect reconstruction filter banks and wavelet bases for \mathbb{R}^n ," *IEEE Trans. Informat. Theory*, vol. 38, pp. 533-555, Mar. 1992.
- [14] T. Lane, Independent JPEG Group's free JPEG software, 1991.
- [15] A. S. Lewis and G. Knowles, "A 64 kb/s video Codec using the 2-D wavelet transform," in *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1991.
- [16] —, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 244-250, Apr. 1992.
- [17] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, pp. 674-693, July 1989.
- [18] —, "Multifrequency channel decompositions of images and wavelet models," *IEEE Trans. Acoust. Speech and Signal Processing*, vol. 37, pp. 2091-2110, Dec. 1990.
- [19] A. Pentland and B. Horowitz, "A practical approach to fractal-based image compression," in *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1991.
- [20] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Mag.*, vol. 8, pp. 14-38, Oct. 1991.
- [21] A. Said and W. A. Pearlman, "Image Compression using the Spatial Orientation Tree," in *Proc. IEEE Int. Symp. Circuits and Syst.*, Chicago, IL, May 1993, pp. 279-282.
- [22] J. M. Shapiro, "An Embedded Wavelet Hierarchical Image Coder," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Francisco, CA, Mar. 1992.
- [23] —, "Adaptive multidimensional perfect reconstruction filter banks using McClellan transformations," *Proc. IEEE Int. Symp. Circuits Syst.*, San Diego, CA, May 1992.
- [24] —, "An embedded hierarchical image coder using zerotrees of wavelet coefficients," in *Proc. Data Compression Conf.*, Snowbird, Utah, IEEE Computer Society Press, 1993.
- [25] —, "Application of the embedded wavelet hierarchical image coder to very low bit rate image coding," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Minneapolis, MN, Apr. 1993.
- [26] Special issue of *IEEE Trans. Informat. Theory*, Mar. 1992.
- [27] G. Strang, "Wavelets and dilation equations: A brief introduction," *SIAM Rev.*, vol. 4, pp. 614-627, Dec. 1989.
- [28] J. Vaisey and A. Geraho, "Image compression with variable block size segmentation," *IEEE Trans. Signal Processing*, vol. 40, pp. 2040-2060, Aug. 1992.
- [29] M. Vetterli, J. Kovačević, and D. J. LeGall, "Perfect reconstruction filter banks for HDTV representation and coding," *Image Commun.*, vol. 2, pp. 349-364, Oct. 1990.
- [30] G. K. Wallace, "The JPEG Still Picture Compression Standard," *Commun. ACM*, vol. 34, pp. 30-44, Apr. 1991.
- [31] I. H. Witten, R. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Comm. ACM*, vol. 30, pp. 520-540, June 1987.
- [32] J. W. Woods, Ed., *Subband Image Coding*. Boston, MA: Kluwer, 1991.
- [33] G. W. Wornell, "A Karhunen-Loève expansion for $1/f$ processes via wavelets," *IEEE Trans. Informat. Theory*, vol. 36, pp. 859-861, July 1990.
- [34] Z. Xiong, N. Galatsanos, and M. Orchard, "Marginal analysis prioritization for image compression based on a hierarchical wavelet decomposition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Minneapolis, MN, Apr. 1993.
- [35] W. Zettler, J. Huffman, and D. C. P. Linden, "Applications of compactly supported wavelets to image compression," *SPIE Image Processing Algorithms*, Santa Clara, CA 1990.



Jerome M. Shapiro (S'85-M'90) was born April 29, 1962 in New York City. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, in 1985, 1985, and 1990, respectively.

From 1982 to 1984, he was at GenRad, Concord, MA, as part of the VI-A Cooperative Program, where he did his Master's thesis on phase-locked loop frequency synthesis. From 1985 to 1987, he was a Research Assistant in the Video Image Processing Group of the MIT Research Laboratory of Electronics. From 1988 to 1990, while pursuing his doctoral studies, he was a Research Assistant in the Sensor Processor Technology Group of MIT Lincoln Laboratory, Lexington, MA. In 1990, he joined the Digital HDTV Research Group of the David Sarnoff Research Center, a Subsidiary of SRI International, Princeton, NJ. His research interests are in the areas of video and image data compression, digital signal processing, adaptive filtering and systolic array algorithms.

Compressing Still and Moving Images with Wavelets *

Michael L. Hilton Björn D. Jawerth Ayan Sengupta

April 18, 1994[†]

Abstract

The wavelet transform has become a cutting-edge technology in image compression research. This article explains what wavelets are and provides a practical, nuts-and-bolts tutorial on wavelet-based compression that will help readers to understand and experiment with this important new technology.

Keywords: image coding, signal compression, wavelet transform, image transforms

1 Introduction

The advent of multimedia computing has led to an increased demand for digital images. The storage and manipulation of these images in their raw form is very expensive; for example, a standard 35mm photograph digitized at 12 μm per pixel requires about 18 MBytes of storage and one second of NTSC-quality color video requires almost 23 MBytes of storage. To make widespread use of digital imagery practical, some form of data compression must be used.

Digital images can be compressed by extracting redundant information. There are three types of redundancy that can be exploited by image compression systems:

- **Spatial Redundancy:** In almost all natural images, the values of neighboring pixels are strongly correlated.
- **Spectral Redundancy:** In images composed of more than one spectral band, the spectral values for the same pixel location are often correlated.
- **Temporal Redundancy:** Adjacent frames in a video sequence often show very little change.

The removal of spatial and spectral redundancies is often accomplished by transform coding, which uses some reversible linear transform to decorrelate the image data (Rabbani and Jones 1991). Temporal redundancy is exploited by techniques that only encode the differences between adjacent frames in the image sequence, such as motion prediction and compensation (Jain and Jain 1981; Liu and Zaccarin 1993).

In the last few years, the wavelet transform has become a cutting edge technology in image compression research. Although the literature on wavelets is vast, most of the papers

*The work in this paper was supported by Summus, Ltd.

[†]To appear in *Multimedia Systems*, Vol. 2, No. 3

dealing with wavelet-based image compression are written by specialists for the specialist. The purpose of this article is to provide a practical, nuts-and-bolts tutorial on wavelet-based compression that will (hopefully) help you to understand and experiment with this important new technology.

This paper is organized into three main sections. Section 2 discusses the theory behind wavelets and why they are useful for image compression. Section 3 describes how the wavelet transform is implemented and used in still image compression systems, and presents some results comparing several different wavelet coding schemes with the JPEG (Wallace 1991) still image compression standard. In Section 4 we describe some initial results with a novel software-only video decompression scheme for the PC environment. We conclude the paper with some remarks about current and future trends in wavelet-based compression.

1.1 A Note on Performance Measures

Throughout this paper, numbers are given for two measures of compression performance — compression ratio and peak signal-to-noise ratio (PSNR). The results of both of these performance measures can be used to mislead the unwary reader, so it is important to explain exactly how these figures were computed. We define compression ratio as

$$\frac{\text{the number of bits in the original image}}{\text{the number of bits in the compressed image}}$$

In this paper we confine our measurements to 8 bits per pixel (bpp) greyscale images, so the peak signal-to-noise ratio in decibels (dB) is computed as

$$\text{PSNR} = 20 \log_{10} \frac{255}{\text{RMSE}}$$

where RMSE is the root mean-squared error defined as

$$\text{RMSE} = \sqrt{\frac{1}{NM} \sum_{i=1}^N \sum_{j=1}^M [f(i, j) - \hat{f}(i, j)]^2}$$

and N and M are the width and height, respectively, of the images in pixels, f is the original image, and \hat{f} is the reconstructed image. Note that the original and the reconstructed images must be the same size.

2 Wavelets

The purpose of this section is to provide an intuitive understanding of what wavelets are and why they are useful for signal compression. For a more rigorous introduction to wavelets, see (Daubechies 1992), (Chui 1992), or (Jawerth and Sweldens 1992).

One of the most commonly used approaches for analyzing a signal $f(x)$ is to represent it as a weighted sum of simple building blocks, called *basis functions*:

$$f(x) = \sum_i c_i \Psi_i(x)$$

where the $\Psi_i(x)$ are basis functions and the c_i are coefficients, or weights. Since the basis functions Ψ_i are fixed, it is the coefficients which contain the information about the signal.

The simplest such representation uses translates of the impulse function as its only bases, yielding a representation that reveals information only about the time domain behavior of the signal. Choosing the sinusoids as the basis functions yields a Fourier representation that reveals information only about the signal's frequency domain behavior.

For the purposes of signal compression, neither of the above representations is ideal. What we would like to have is a representation which contains information about both the time and frequency behavior of the signal. More specifically, we want to know the frequency content of the signal at a particular instant in time. However, resolution in time (Δx) and resolution in frequency ($\Delta \omega$) cannot both be made arbitrarily small at the same time because their product is lower bounded by the Heisenberg inequality

$$\Delta x \Delta \omega \geq \frac{1}{2}.$$

This inequality means that we must trade off time resolution for frequency resolution, or vice versa. Thus, it is possible to get very good resolution in time if you are willing to settle for low resolution in frequency, and you can get very good resolution in frequency if you are willing to settle for low resolution in time.

The situation is really not all that bad from a compression standpoint. By their very nature, low frequency events are spread out (or non-local) in time and high frequency events are concentrated (or localized) in time. Thus, one way that we can live within the confines of the Heisenberg inequality and yet still get useful time-frequency information about a signal is if we design our basis functions to act like cascaded octave bandpass filters, which repeatedly split the signal's bandwidth in half.

To gain insight into designing a set of basis functions that will satisfy both our desire for information and the Heisenberg inequality, let us compare the impulse function and the sinusoids. The impulse function cannot provide information about the frequency behavior of a signal because its support — the interval over which it is non-zero — is infinitesimally small. At the opposite extreme are the sinusoids, which cannot provide information about the time behavior of a signal because they have infinite support. ~~What we need, then, is a compromise between these two extremes: a set of basis functions $\{\Psi_i\}$, each with finite support, of a different width.~~ The different support widths allow us to trade off time and frequency resolution in different ways; for example, a wide basis function can examine a large region of the signal and resolve low frequency details accurately, while a short basis function can examine a small region of the signal to resolve time details accurately.

To simplify things, let us constrain all of the basis functions in $\{\Psi_i\}$ to be scaled and translated versions of the same prototype function Ψ , known as the *mother wavelet*. The scaling is accomplished by multiplying x by some scale factor; if we choose the scale factor to be a power of 2, yielding $\Psi(2^\nu x)$ where ν is some integer, we get the cascaded octave bandpass filter structure we desire. Because Ψ has finite support, it will need to be translated along the time axis in order to cover an entire signal. This translation is accomplished by considering all the integral shifts of Ψ ,

$$\Psi(2^\nu x - k), \quad k \in \mathcal{Z}.$$

Note that this really means that we are translating Ψ in steps of size $2^{-\nu}k$.¹ Putting this all together gives us a *wavelet decomposition* of the signal,

$$f(x) = \sum_{\nu \text{ finite}} \sum_{k \text{ finite}} c_{\nu k} \Psi_{\nu k}(x)$$

¹This is because $\Psi(2^\nu x - k) = \Psi(2^\nu(x - 2^{-\nu}k))$.

where

$$\Psi_{\nu k}(x) = 2^{\nu/2} \Psi(2^{\nu}x - k)$$

(the multiplication by $2^{\nu/2}$ is needed to make the bases orthonormal). So far we have said nothing about the coefficients $c_{\nu k}$. They are computed by the *wavelet transform*, which is just the inner product of the signal $f(x)$ with the basis functions $\Psi_{\nu k}(x)$.

The comparisons between wavelets and octave bandpass filters was not made just for pedagogical reasons. Wavelets can, in fact, be thought of and implemented as octave bandpass filters, and we shall treat them as such for the remainder of this paper.

3 Still Image Compression

A wide variety of wavelet-based image compression schemes have been reported in the literature, ranging from simple entropy coding to more complex techniques such as vector quantization (Antonini et al. 1992; Hopper and Preston 1992), adaptive transforms (Desarte et al. 1992; Wickerhouser 1992), tree encodings (Shapiro 1993; Lewis and Knowles 1992), and edge-based coding (Froment and Mallat 1992). All of these schemes can be described in terms of the general framework depicted in Fig. 1. Compression is accomplished by applying a wavelet transform to decorrelate the image data, quantizing the resulting transform coefficients, and coding the quantized values. Image reconstruction is accomplished by inverting the compression operations. We now describe each of the boxes in Fig. 1 in more detail.

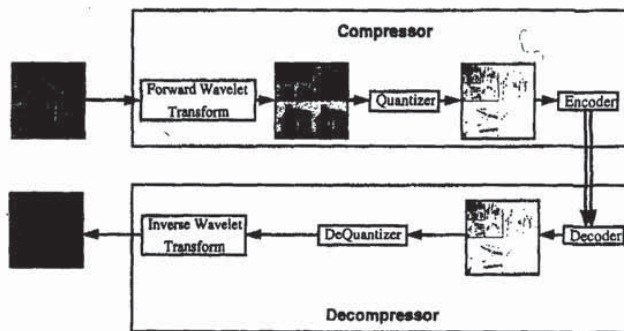


Figure 1: Block diagram of wavelet-based image coders.

3.1 Implementing the Wavelet Transform

The forward and inverse wavelet transforms can each be efficiently implemented in $\mathcal{O}(n)$ time by a pair of appropriately designed Quadrature Mirror Filters (QMFs) (Croisier et al. 1976). Therefore, wavelet-based image compression can be viewed as a form of subband coding (Woods and O'Neil 1986). Each QMF pair consists of a lowpass filter (H) and a highpass filter (G) which split a signal's bandwidth in half. The impulse responses of H and G are mirror images, and are related by

$$g_n = (-1)^{1-n} h_{1-n}. \quad (1)$$

The impulse responses of the forward and inverse transform QMFs — denoted (\tilde{H}, \tilde{G}) and (H, G) respectively — are related by

$$g_n = \tilde{g}_{-n} \quad (2)$$

$$h_n = \tilde{h}_{-n}. \quad (3)$$

To illustrate how the wavelet transform is implemented, we shall use Daubechie's \mathcal{W}_6 wavelet (Daubechie 1988). We chose this wavelet because it is well known and has some nice properties. One such property is that it has two vanishing moments, which means the transform coefficients will be zero (close to zero) for any signal that can be described by (approximated by) a polynomial of degree 2 or less. The mother wavelet basis for \mathcal{W}_6 is shown in Fig. 2. The filter coefficients for H of \mathcal{W}_6 are

$$\begin{aligned} h_0 &= 0.332670552950 \\ h_1 &= 0.806891509311 \\ h_2 &= 0.459877502118 \\ h_3 &= -0.135011020010 \\ h_4 &= -0.085441273882 \\ h_5 &= 0.035226291882 \end{aligned}$$

from which the coefficients for G , \tilde{H} , and \tilde{G} can be derived using Equations 1, 2, and 3. The impulse responses of H and G are shown in Fig. 3.

A one-dimensional signal s can be filtered by convolving the filter coefficients c_k with the signal values:

$$\hat{s}_i = \sum_{k=0}^M c_k s_{i-k}$$

where M is the number of coefficients, or *taps*, in the filter. The one-dimensional forward wavelet transform of a signal s is performed by convolving s with both \tilde{H} and \tilde{G} and downsampling by 2. As dictated by Equation 1, the relationship of the \tilde{H} and \tilde{G} filter coefficients with the beginning of signal s is

$$\begin{array}{cccccccc} & \tilde{h}_5 & \tilde{h}_4 & \tilde{h}_3 & \tilde{h}_2 & \tilde{h}_1 & \tilde{h}_0 & & \\ & s_0 & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & \dots \\ \tilde{g}_5 & \tilde{g}_4 & \tilde{g}_3 & \tilde{g}_2 & \tilde{g}_1 & \tilde{g}_0 & & & \end{array}$$

Note that the \tilde{G} filter extends before the signal in time; if s is finite, the \tilde{H} filter will extend beyond the end of the signal. A similar situation is encountered with the inverse wavelet transform filters H and G . In an implementation, one must make some choice about what

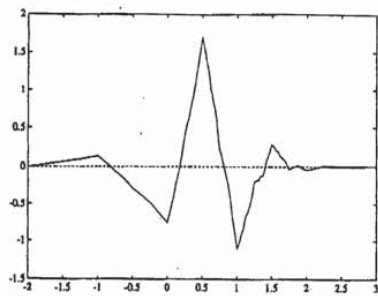


Figure 2: The mother wavelet basis function for W_6 .

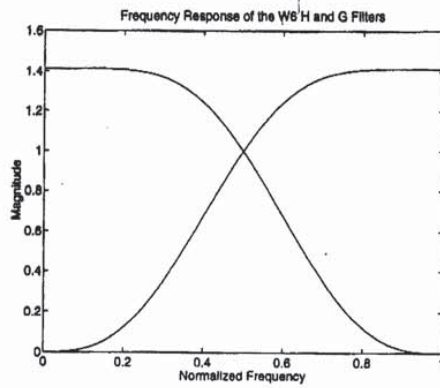


Figure 3: Frequency response of the W_6 QMFs.

values to pad the extensions with. A choice which works well in practice is to wrap the signal about its endpoints, i.e.,

$$\dots s_{n-1} s_n \boxed{s_0 s_1 s_2 \dots s_{n-2} s_{n-1} s_n} s_0 s_1 \dots,$$

thereby creating a periodic extension of s .

Fig. 4 illustrates a single 2-D forward wavelet transform of an image, which is accomplished by two separate 1-D transforms. The image $f(x, y)$ is first filtered along the x dimension, resulting in a lowpass image $f_L(x, y)$ and a highpass image $f_H(x, y)$. Since the bandwidth of f_L and f_H along the x dimension is now half that of f , we can safely downsample each of the filtered images in the x dimension by 2 without loss of information. The downsampling is accomplished by dropping every other filtered value. Both f_L and f_H are then filtered along the y dimension, resulting in four subimages: f_{LL} , f_{LH} , f_{HL} , and f_{HH} . Once again, we can downsample the subimages by 2, this time along the y dimension. As illustrated in Fig. 4, the 2-D filtering decomposes an image into an *average signal* (f_{LL}) and three *detail signals* which are directionally sensitive: f_{LH} emphasizes the horizontal image features, f_{HL} the vertical features, and f_{HH} the diagonal features. The directional sensitivity of the detail signals is an artifact of the frequency ranges they contain.

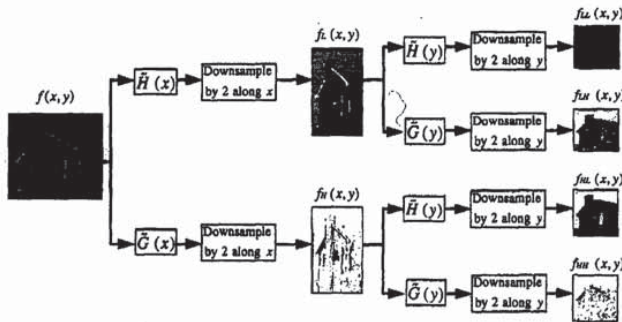


Figure 4: Block diagram of the 2-D forward wavelet transform.

It is customary in wavelet compression to recursively transform the average signal.² The number of transformations performed depends on several factors, including the amount of compression desired, the size of the original image, and the length of the QMF filters. In general, the higher the desired compression ratio, the more times the transform is performed.

After the forward wavelet transform is completed, we are left with a matrix of coefficients that comprise the average signal and the detail signals of each scale. No compression of the original image has been accomplished yet; in fact, each application of the forward wavelet

²A more sophisticated decomposition strategy is to use the *wavelet packets* of Coifman and Meyer (Wickerhouser 1992; Coifman and Wickerhouser 1992).

transform causes the magnitude of the coefficients to grow, so there has actually been an increase in the amount of storage required for the image! Compression is achieved by quantizing and encoding the wavelet coefficients.

The 2-D inverse wavelet transform is illustrated in Fig. 5. The average and detail signals are first upsampled by 2 along the y dimension. Upsampling is accomplished by inserting a zero between each pair of values in the y dimension. The upsampling is necessary to recover the proper bandwidth required to add the signals back together. After upsampling, the signals are filtered along the y dimension and added together appropriately. The process is then repeated in the x dimension, yielding the final reconstructed image.

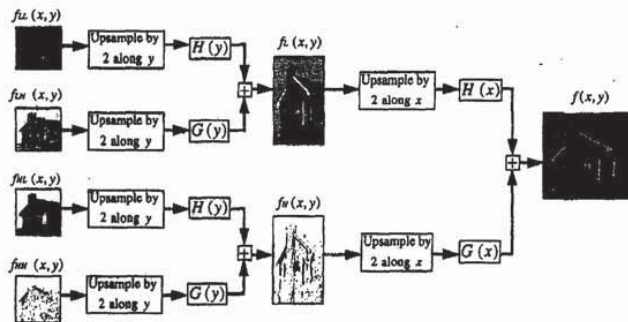


Figure 5: Block diagram of the 2-D inverse wavelet transform.

3.2 Quantization

The forward wavelet transform decorrelates the pixel values of the original image and concentrates the image information into a relatively small number of coefficients. Fig. 6 (Left) is a histogram of the pixel values for the 8-bits per pixel (bpp) 512×512 Lena image, and Fig. 6 (Right) is a histogram of the wavelet coefficients of the same image after the forward wavelet transform is applied. The "information packing" effect of the wavelet transform is readily apparent from the scarcity of coefficients with large magnitudes.

The sharply peaked coefficient distribution of the wavelet transformed image has a lower zero-th order entropy (4.24 bpp) than the original image (7.46 bpp), thereby increasing the amount of lossless compression possible.

We can also take advantage of the energy invariance property of the wavelet transform to achieve high-quality lossy compression. The energy invariance property says that total amount of energy in an image does not change when the wavelet transform is applied. This property can also be viewed in a slightly different way: any changes made to the values of the wavelet coefficients will result in proportional changes in the pixel values of the reconstructed image. In other words, we can eliminate (set to zero) those coefficients with small magnitudes without creating significant distortion in the reconstructed image. In

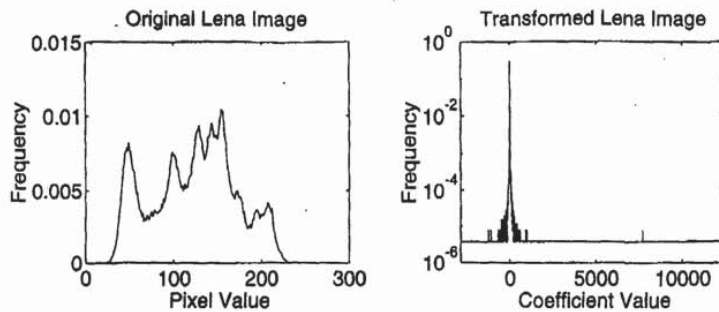


Figure 6: LEFT) Normalized histogram of the pixel values in the original Lena image. RIGHT) Normalized histogram of the wavelet transform coefficients of the same image.

practice, it is possible to eliminate all but a few percent of the wavelet coefficients and still get a reconstructed image of reasonable quality. The elimination of small valued coefficients can be accomplished by applying a *thresholding function*

$$T(t, x) = \begin{cases} 0 & \text{if } |x| < t \\ x & \text{otherwise} \end{cases}$$

to the coefficient matrix. The amount of compression obtained can now be controlled by varying the threshold parameter t .

Higher compression ratios can be obtained by *quantizing* the non-zero wavelet coefficients before they are encoded. A quantizer is a many-to-one function $Q(x)$ that maps many input values into a (usually much) smaller set of output values. Quantizers are staircase functions characterized by a set of numbers $\{d_i, i = 0, \dots, N\}$ called *decision points* and a set of numbers $\{r_i, i = 0, \dots, N - 1\}$ called *reconstruction levels*. An input value x is mapped to a reconstruction level r_i if x lies in the interval $(d_i, d_{i+1}]$.

To achieve the best results, a separate quantizer should be designed for each scale, taking into account both the properties of the Human Visual System (Marr 1982) and the statistical properties of the scale's coefficients. The characteristics of the Human Visual System guide the allocation of bits among the different scales, and the coefficient statistics guide the quantizer design for each scale. Descriptions of various bit allocation strategies can be found in (Matic and Mosley 1993) and (Clarke 1985).

The distribution of coefficient values in the various detail signals can be modeled reasonably well by the Generalized Gaussian Distribution (GGD). The probability density function of the coefficient distribution at each scale ν , then, can be given by (Abramowitz and Stegun 1965):

$$p_\nu(x) = \left[\frac{\alpha_\nu \eta(\alpha_\nu, \sigma_\nu)}{2\Gamma(1/\alpha_\nu)} \right] \exp(-[\eta(\alpha_\nu, \sigma_\nu) |x|]^{\alpha_\nu})$$

Scale ν	Codeword Size (in bits)	Decision Points and Reconstruction Levels	
8	2	d_i	5, 10, 23, 48, 256
		r_i	7, 15, 32, 65
7	3	d_i	5, 10, 18, 28, 40, 57, 81, 117, 512
		r_i	7, 14, 22, 33, 47, 67, 95, 139
6	3	d_i	10, 16, 26, 41, 63, 95, 144, 223, 1024
		r_i	12, 20, 32, 50, 76, 114, 174, 271
5	5	d_i	20, 33, 51, 73, 99, 128, 161, 201, 245 291, 339, 386, 436, 500, 591, 738, 2048
		r_i	25, 41, 61, 85, 113, 143, 179, 223, 267
			314, 362, 410, 461, 539, 644, 834

Table 1: Lloyd-Max quantizers generated using magnitude data from the \mathcal{W}_8 transformed Lena image.

where

$$\eta(\alpha, \sigma) = \sigma^{-1} \left[\frac{\Gamma(3/\alpha)}{\Gamma(1/\alpha)} \right]^{1/2}$$

and

$$\Gamma(a) = \int_0^{\infty} e^{-t} t^{a-1} dt.$$

σ_ν is the standard deviation of the coefficient distribution at scale ν and α_ν is a shape parameter describing the exponential rate of decay of the distribution at scale ν . For example, when $\alpha_\nu = 1$ the GGD becomes the Laplacian pdf, while $\alpha_\nu = 2$ leads to the Gaussian pdf. The α_ν appropriate for a particular class of images can be computed using the χ^2 test or by simple observation. For the \mathcal{W}_8 wavelet and the Lena image, several of the appropriate values of α_ν and σ_ν are:

$$\begin{array}{ll} \alpha_8 = 0.58 & \sigma_8 = 5.48 \\ \alpha_7 = 0.49 & \sigma_7 = 9.39 \\ \alpha_6 = 0.43 & \sigma_6 = 14.33 \\ \alpha_5 = 0.39 & \sigma_5 = 20.17 \end{array}$$

The design of scalar quantizers will also depend on the type of encoder to be used. If the encoder uses fixed-length codewords, the Lloyd-Max algorithm (Max 1960) can be used to design a quantizer that minimizes the mean-squared quantization error. If a variable-length entropy coder is used, uniform quantization is optimal (in the mean-squared error sense) when the coefficient distribution is Laplacian; for other distributions, the algorithm in (Wood 1969) can be used to design an optimal quantizer. Vector quantization (Gersho and Grey 1992) has also been used in wavelet compression systems, for example (Antonini et al. 1992) and (Bradley and Brislawn 1993).

Table 1 lists the decision points and reconstruction levels for a set of Lloyd-Max quantizers generated using magnitude data from the \mathcal{W}_8 transformed Lena image. One extra bit per codeword is needed to represent the sign of the quantized coefficient. The codeword sizes were chosen by experimentation.

3.3 Coding the Coefficients

The encoder/decoder pair, or *codec*, has the task of losslessly compressing and decompressing the sparse matrix of quantized coefficients. Codec design has received a tremendous amount of attention, and a wide variety of schemes exist (Lelewer and Hirshberg 1987). The design of a codec is usually a compromise between (often conflicting) requirements for memory use, execution speed, available bandwidth, and reconstructed image quality.

For applications requiring fast execution, simple run-length coding (Pratt 1978) of the zero-valued coefficients has proven very effective. (The distribution of non-zero coefficients is such that rarely is it profitable to run-length encode them.) The zero run-lengths can be encoded using either fixed-length codewords or variable-length entropy coding; entropy coding is more expensive to implement, but can improve the peak signal to noise ratio (PSNR) of reconstructed images by as much as 3 dB, depending upon the energy-packing ability of the wavelet in use.

For applications requiring the best possible image quality at a particular compression ratio, a technique such as Shapiro's Zero Tree encoding (Shapiro 1993) is a better choice. The execution speed tradeoff between these two codecs is quite dramatic: our run-length entropy coder takes less than one second to compress a 512×512 8 bpp image on a 66-MHz 80486 computer, and Zero Tree-like coders can take up to 45 seconds to compress the same image on the same machine. However, the quality of the Zero Tree image is much better — 36.28 dB PSNR (Shapiro 1993) vs. 33.2 dB PSNR at a compression ratio of 16:1.

3.4 Compression Results

The peak signal to noise ratios of several different wavelet compression techniques applied to the 512×512 8-bpp Lena image are compared in Fig. 7. The graphs show that both the encoding technique and the particular wavelet used can make a significant difference in the performance of a compression system: the Zerotree coder performs the best; biorthogonal wavelets (Antonini et al. 1992; Cohen 1992; Averbuch et al. 1993) perform better than \mathcal{W}_6 ; and variable length coders perform better than fixed length coders.

The performance of a baseline JPEG (Wallace 1991) image compressor³ is also indicated in Fig. 7. At compression ratios less than 25:1 or so, JPEG performs better numerically than the simple wavelet coders. At compression ratios above 30:1, JPEG performance rapidly deteriorates, while wavelet coders degrade gracefully well beyond ratios of 100:1. Figure 8 compares the visual quality of several image coders.

4 Video Compression

The wavelet transform can also be used in the compression of image sequences, or video. Video compression techniques are able to achieve high quality image reconstruction at low bit rates by exploiting the temporal redundancies present in an image sequence (Le Gall 1991; Liu 1991). Wavelet-based implementations of at least two standard video compression techniques, hierarchical motion compensation (Uz et al. 1991) and 3-D subband coding (Karlson and Vetterli 1989), have been reported (Zhang and Zafar 1992; Lewis and Knowles 1990). However, the computational expense of the wavelet transform has so far prevented its use in realtime, software-only video codecs for PC-class computers. In this section, we

³The JPEG coder that is included in Version 2.21 of John Bradley's *xview* program was used to generate the JPEG performance data shown in Fig. 7.

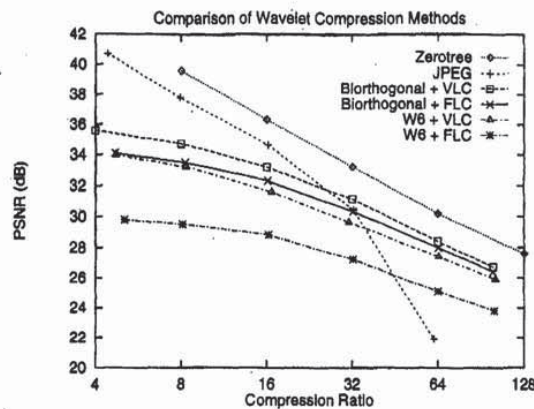


Figure 7: A comparison of the image reconstruction quality of several different wavelet coders and JPEG. The tests were performed on the 512×512 , 8-bpp Lena image. "VLC" means Variable Length Coder, and "FLC" means Fixed Length Coder.

describe a new technique for rapidly evaluating the inverse wavelet transform and illustrate its use in the context of a software-only video decoder based on frame differencing.

4.1 The Basic Idea

The playback speed of a wavelet-based video coder depends in large part upon how long it takes to perform the inverse wavelet transform. A 66-Mhz 80486 computer takes about 0.25 seconds to compute a complete inverse wavelet transform for a 256×256 , 8-bpp greyscale image. Unless one finds a way to avoid performing a complete inverse transform each time an image frame is reconstructed, wavelets are not viable for software-only video of reasonably sized images.

Fortunately, it is not necessary to perform the complete inverse transform for each frame in a slowly varying image sequence. The value of an arbitrary pixel p in an image is determined by a weighted sum of all the basis vectors in the wavelet decomposition that include p in their region of support. If the weights (i.e., the wavelet coefficients) of these basis vectors do not change between frames in an image sequence, then the value of pixel p will not change either. Therefore, it is not necessary to compute the inverse wavelet transform for those regions of the image that have not changed between frames. This idea was first put forth in (Andersson et al. to appear).

The basic idea for rapidly decompressing image sequences, then, is to only compute the inverse wavelet transform for those pixels influenced by coefficients that have change by a meaningful amount between adjacent frames in the sequence.

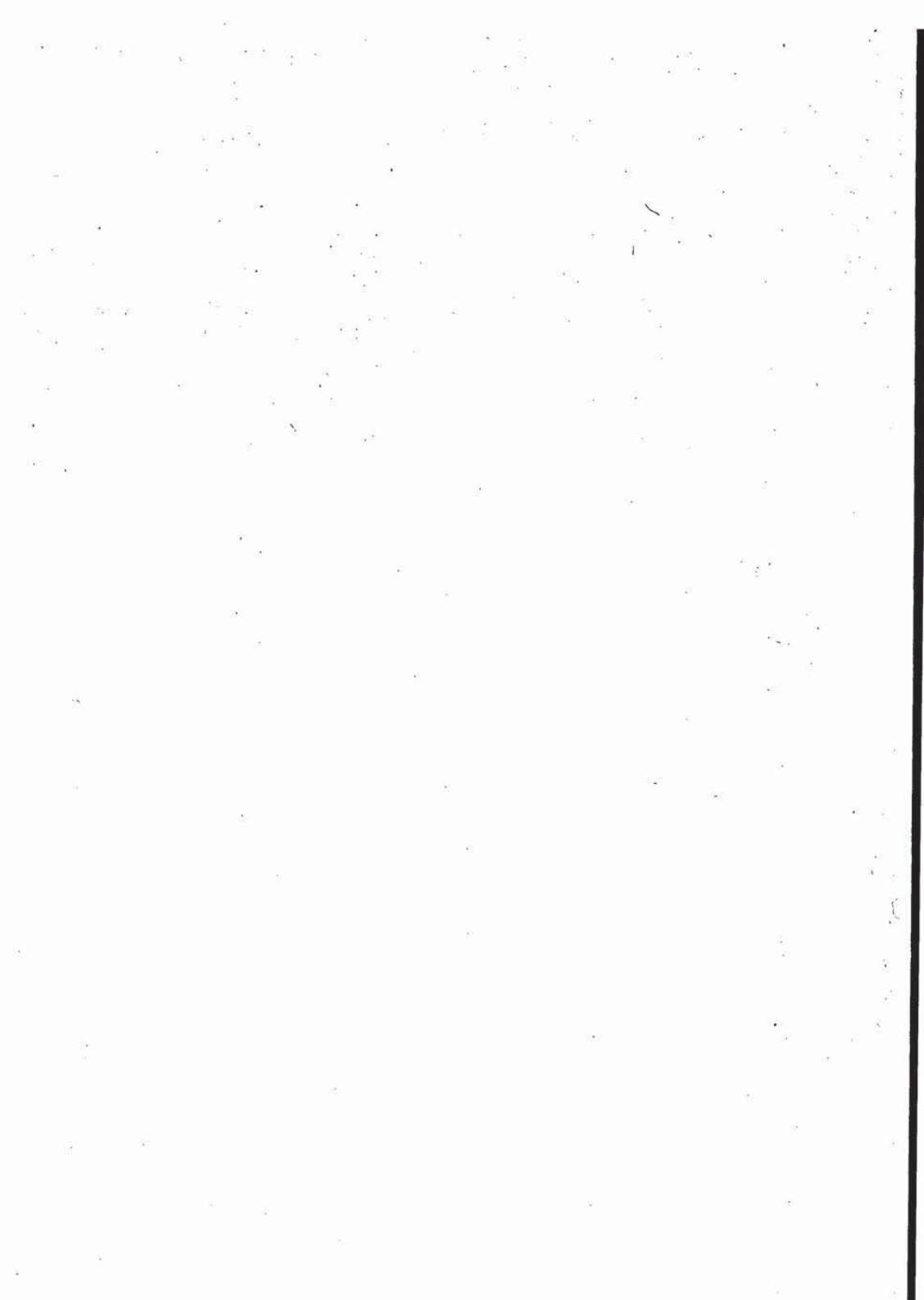


Figure 8: Reconstructed images for the \mathcal{W}_6 +VLC, Biorthogonal+VLC, and JPEG image coders.

4.2 A Simple Frame Differencing Video Coder

The simple video coder described herein is shown in Figs. 9 and 10. Let us consider a sequence of images $\{f_i\}_{i=0,1,\dots}$, where each f_i denotes the i th frame in the sequence. The difference between two adjacent frames is given by

$$\Delta f_i = f_{i+1} - f_i.$$

Δf_i is called a *difference image*, and it contains only the change in image content between frame f_i and f_{i+1} — it does not contain any redundant first-order temporal information. There is spatial redundancy in Δf_i , however, and this redundancy can be reduced by application of some wavelet transform W .⁴ Thresholding can now be performed on the transformed difference image $W(\Delta f_i)$ to eliminate image changes that are considered too small to be meaningful. After thresholding, we now have an approximate transformed difference image $\widehat{W}(\Delta f_i)$ that is extremely sparse. $\widehat{W}(\Delta f_i)$ is then analyzed to determine which portions of the inverse wavelet transform will need to be performed to reconstruct an approximation $\widehat{\Delta f}_i$ of the i th difference image. This information is then encoded and sent to the video decoder.

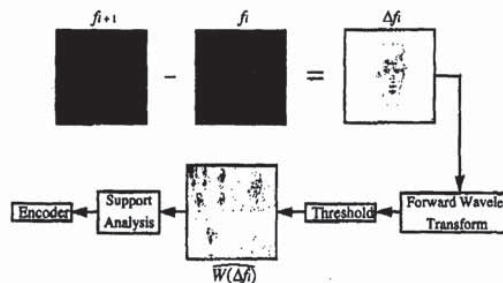


Figure 9: Block diagram of the video encoder.

Using the information sent by the encoder, the decoder can reconstruct $\widehat{\Delta f}_i$. Because of its sparse nature, $\widehat{\Delta f}_i$ can be reconstructed very quickly by computing the inverse wavelet transform for only those pixels influenced by the coefficients sent by the encoder. We assume that the decoder has available some approximation \hat{f}_i of frame i , so the next frame in the sequence can be constructed as

$$\hat{f}_{i+1} = \hat{f}_i + \widehat{\Delta f}_i.$$

We have implemented a prototype video compression system based on the ideas described above and achieved promising initial results. The results of an experiment in which we compressed 30 frames of the standard Miss America video sequence (the images were first rescaled to 256×256 pixels) are presented in Table 2. The experiment was performed on a 66 MHz 80486 computer running the OS/2 operating system, and the entire video

⁴We note that because the wavelet transform is linear, it does not matter from a theoretical standpoint if we form $W(\Delta f_i)$ by $W(f_{i+1}) - W(f_i)$ or by $W(f_{i+1} - f_i)$.

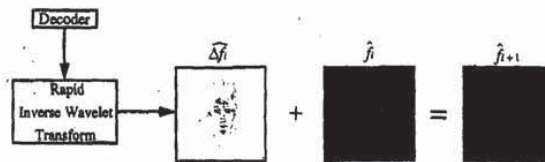


Figure 10: Block diagram of the video decoder.

Threshold	Compressed Size (Bytes)	Compression Ratio	Transform (sec)	Decompression (sec)	Speed (fps)	PSNR (dB)
0	172,2278	11:1	0.074	0.157	6.4	40.73
10	150,977	13:1	0.073	0.150	6.6	38.03
20	107,351	18:1	0.061	0.121	8.3	35.06
30	91,365	22:1	0.056	0.115	8.6	32.22

Table 2: Results of Compression experiments on 30 frames of the Miss America video sequence. The original, uncompressed sequence requires 1,966,080 bytes of storage. All times and PSNRs are mean values for the entire sequence.

coder and decoder are written in the C programming language. The wavelet we use is an integerized version of Daubechies' W_6 .

Performing the complete 2-dimensional inverse wavelet transform takes 0.25 seconds per frame. Our experiments indicate that at compression ratios of around 20:1, the partial 2-dimensional inverse transform technique is more than four times faster than the full inverse transform, and is capable of transforming over 16 image frames per second. This measurement is only of the time required to perform the inverse wavelet transform — it does not include the time it takes to decode the image data or display the reconstructed image.

Our codec currently uses a combination of fixed and variable length codes for representing the video data. Our primary concern so far has been increasing the speed of the inverse wavelet transform, and we have not paid much attention to coding issues. The development of codes which can be quickly decoded is of major importance, because the time required for decoding the compressed image data is presently the performance bottleneck of our experimental decompression system.

5 Concluding Remarks

Basic and applied research in the field of wavelets has made tremendous progress in the last five years. Image compression schemes based on wavelets are rapidly gaining maturity, and have already begun to appear in commercial software/hardware systems. The reconstruction quality of wavelet compressed images has already moved well beyond capabilities of JPEG, which is the current international standard for image compression.

Video is the next big challenge for wavelet-based data compression. Our laboratory

experiments using three dimensional wavelet transforms to compress a $64 \times 64 \times 64$ color video sequence indicate that visually lossless video compression is possible at compression rates near 1000:1, but the memory and processor requirements are presently too great to make such a scheme practical. The technique presented in Section 4 of this paper is a small step towards a practical video compression scheme, but much research remains to be done.

It is also interesting to note that wavelet research on image compression has had a strong impact on several areas of numerical analysis, especially in the solution of partial differential equations (Alpert 1992; Alpert et al. 1993; Beylkin et al. 1991). The compression of an image, which is just a matrix of intensity values, is not really different from compressing the kernel matrix of a functional operator. The compressed operator is a sparse matrix, and sparse matrix operations can often be performed orders of magnitude faster than their non-sparse counterparts. Undoubtedly, this will lead to new results in numerical analysis that will impact image compression, leading to better algorithms in areas such as computer vision.

References

Abramowitz M, Stegun IA (1965) Handbook of Mathematical Functions. Dover, New York.

Alpert B, Beylkin G, Coifman R, Rokhlin V (1993) Wavelet-like bases for the fast solution of second-kind integral equations. *SIAM Journal of Scientific Computing*, 14(1):159-184.

Alpert B (1992) Wavelets and other bases for fast numerical linear algebra. In: Chui CK (ed) *Wavelets: A Tutorial in Theory and Applications*, Volume 2. Academic Press, San Diego, pp. 181-216.

Andersson L, Hall N, Jawerth B, Peters G (1994) Wavelets on closed subsets of the real line. In: Schumacher LL, Webb G (ed) *Recent Advances in Wavelet Analysis*. Academic Press, San Diego, pp. 1-61.

Antonini M, Barlaud M, Mathieu P, Daubechies I (1992) Image coding using wavelet transform. *IEEE Trans Image Processing*, 1(2):205-220.

Averbuch A, Lazar D, Israeli M (1993) Image compression using wavelet transform and multiresolution decomposition. In: Storer JA, Cohn M (ed) *Proceedings Data Compression Conference 93*. IEEE Computer Society Press, p. 459.

Beylkin G, Coifman R, Rokhlin V (1991) Fast wavelet transforms and numerical algorithms I. *Commun Pure and Applied Math*, 44:141-183.

Bradley JN, Brislawn CM (1993) Wavelet transform-vector quantizer compression of supercomputer ocean models. In: Storer JA, Cohn M (ed) *Proceedings Data Compression Conference 93*. IEEE Computer Society Press, pp. 224-233.

Chui CK (1992) An Introduction to Wavelets, volume 1 of Wavelet Analysis and Its Applications. Academic Press, San Diego.

Clarke RJ (1985) Transform Coding of Images. Academic Press, San Diego.

Cohen A (1992) Biorthogonal wavelets. In: Chui CK (ed) Wavelets: A Tutorial in Theory and Applications, Volume 2. Academic Press, San Diego, pp. 123-152.

Coifman R, Wickerhauser MV (1992) Entropy-based algorithms for best basis selection. IEEE Trans Information Theory, 38(2):713-718, Part II.

Croisier A, Esteban D, Galand C (1976) Perfect channel splitting by use of interpolation/decimation tree decomposition techniques. In: Int Conf on Information Science and Systems.

Daubechies I (1988) Orthonormal bases of compactly supported wavelets. Commun Pure and Applied Math, 41:909-996.

Daubechies I (1992) Ten Lectures on Wavelets, volume 61 of CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM.

Desarte P, Macq B, Slock DTM (1992) Signal-adapted multiresolution transform for image coding. IEEE Trans Information Theory, 38(2):897-904.

Froment J, Mallat S (1992) Second generation compact image coding with wavelets. In: Chui CK (ed) Wavelets: A Tutorial in Theory and Applications, Volume 2. Academic Press, San Diego, pp. 655-678.

Le Gall D (1991) MPEG: A video compression standard for multimedia applications. Commun ACM, 34(4):46-58.

Gersho A, Gray RM (1992) Vector Quantization and Signal Compression. Kluwer Academic Publishers, Boston Dordrecht London.

Hopper T, Preston F (1992) Compression of grey-scale fingerprint images. In: Storer JA, Cohn M (ed) Proceedings Data Compression Conference 92. IEEE Computer Society Press, pp. 309-318.

Jain JR, Jain AK (1981) Displacement measurement and its application in interframe image coding. IEEE Trans Communications, COM-29(12):1799-1808.

Jawerth B, Sweldens W (1992) An overview of wavelet based multiresolution analyses.

Technical report, Industrial Mathematics Initiative, The University of South Carolina Department of Mathematics.

Karlsson G, Vetterli M (1989) Packet video and its integration into the network architecture. *IEEE J Selected Areas in Communications*, 7(5):739-751.

Lelewer DA, Hirshberg DS (1987) Data compression. *ACM Computing Surveys*, 19(3):261-295.

Lewis AS, Knowles G (1990) Video compression using 3D wavelet transforms. *Electronics Letters*, 26(6):396-397.

Lewis AS, Knowles G (1992) Image compression using the 2-d wavelet transform. *IEEE Trans Image Processing*, 1(2):244-250.

Liu B, Zaccarin A (1993) New fast algorithms for the estimation of block motion vectors. *IEEE Trans Circuits Systems for Video Technology*, 3(2):148-157.

Liu M (1991) Overview of the px64 kbit/s video coding standard. *Commun ACM*, 34(4):59-63.

Marr D (1982) *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman.

Matic R, Mosley J (1993) Wavelet transform-adaptive scalar quantization of multispectral data. In: *AIAA Computing in Aerospace 9 Conference*. American Institute of Aeronautics and Astronautics, addendum 93-4485.

Max J (1960) Quantizing for minimum distortion. *IRE Trans on Information Theory*, IT-6(1):7-12.

Pratt WK (1978) *Digital Image Processing*. Wiley, New York.

Rabbani M, Jones P (1991) *Digital Image Compression Techniques*, volume TT7 of SPIE Tutorial Texts in Optical Engineering. SPIE Press.

Shapiro JM (1993) Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans Signal Processing*, 41(12):3445-3462.

Uz KM, Vetterli M, Le Gall D (1991) Interpolative multiresolution coding of advanced television with compatible subchannels. *IEEE Trans Circuits Systems for Video Technology*, 1(1):86-99.










Wallace GK (1991) The JPEG still picture compression standard. Commun ACM, 34(4):30-44.

Wickerhauser MV (1992) Acoustic signal compression with wavelet packets. In: Chui CK (ed) Wavelets: A Tutorial in Theory and Applications, Volume 2. Academic Press, San Diego, pp. 679-700.

Wood RC (1969) On optimum quantization. IEEE Trans Information Theory, IT-15(2):248-252.

Woods JW, O'Neil SD (1986) Subband coding of images. IEEE Trans Acoustics, Speech, Signal Proc, ASSP-34(5):1278-1288.

Zhang Y, Zafar S (1992) Motion-compensated wavelet transform coding for color video compression. IEEE Trans Circuits Systems for Video Technology, 2(3):285-296.

	W_6+VLC	Biorth+VLC	JPEG
16:1			
32:1			
64:1			

On the Modeling of DCT and Subband Image Data for Compression

Keith A. Birney and Thomas R. Fischer, Senior Member, IEEE

Abstract—Image subband and discrete cosine transform coefficients are modeled for efficient quantization and noiseless coding. Quantizers and codes are selected based on Laplacian, fixed generalized Gaussian, and adaptive generalized Gaussian models. The quantizers and codes based on the adaptive generalized Gaussian models are always superior in mean-squared error distortion performance but, generally, by no more than 0.08 b/pixel, compared with the much simpler Laplacian model-based quantizers and noiseless codes. This provides strong motivation for the selection of pyramid codes for transform and subband image coding.

I. INTRODUCTION

EFFECTIVE quantization and noiseless coding are dependent on good source models. In discrete cosine transform (DCT) image coding, Reiningger and Gibson [1] use the Kolmogorov-Smirnov goodness-of-fit tests in order to conclude that the non-dc transform coefficients can be better modeled as Laplacian than as Gaussian, Rayleigh, or gamma distributed. Tanabe and Farvardin [2] model image subband and DCT coefficients using the generalized Gaussian density

$$p(x) = \frac{\nu \eta(\nu, \sigma)}{2\Gamma(1/\nu)} \exp(-[\eta(\nu, \sigma)|x|]^\nu) \quad (1)$$

where

$$\eta(\nu, \sigma) = \frac{1}{\sigma} \left[\frac{\Gamma(3/\nu)}{\Gamma(1/\nu)} \right]^{1/2} \quad (2)$$

and conclude that parameter value $\nu = 0.7$ is most representative of image subband data, whereas $\nu = 0.8$ is the best choice for non-dc DCT coefficients, given an 8×8 DCT applied to the lowest frequency subband. (A generalized Gaussian parameter value of 1.0 for ν yields the Laplacian density, whereas a value of 2.0 yields the Gaussian density.)

The basic goal of this paper is to further quantify the results of [1] and [2] by studying the consequences of the modeling assumptions on the resulting coding performance. In Section II, we define a system in which each respective DCT coefficient or subband is modeled by a geometric distribution and then encoded with the optimum entropy constrained

uniform threshold quantizer (UTQ) [3] for that source model. (Due to the entropy constraint, the quantizer may be considered to have a number of output levels sufficient to prevent clipping.) The UTQ entropy versus distortion (mean square error) performance is essentially the same as for the optimum entropy constrained scalar quantizer for the source model under consideration, and it never differs from the source rate distortion function by more than 0.255 b/sample [3]. Optimum rate allocation is applied such that the overall mean squared error is minimized. We compare the peak signal-to-noise ratio (PSNR) versus rate (entropy) performance of adaptive generalized Gaussian model-based quantization with that of simpler Laplacian and fixed generalized Gaussian model-based quantization. Assuming that the respective quantized DCT coefficients or subbands can be noiselessly encoded with zero redundancy (that is, at their first-order entropy), we make the empirical observation that the more complex adaptive generalized Gaussian modeling offers virtually no improvement in PSNR performance.

In Section III, we extend the coding system to include a model-based noiseless encoding in order to compare the applicability of the three modeling approaches to such an application. That is, for each DCT coefficient or subband, we let the geometric distribution model define an ideal noiseless code and then determine the redundancy of the code with respect to the distribution of the quantized data, as measured by the discrimination [4].

Results for the model-based quantization and noiseless coding system are given in Section IV. The general conclusion is that for a wide class of images, the simpler Laplacian model-based coding is quite robust. The generalized Gaussian model-based coding is always superior but by no more than 0.08 b/pixel. Very little is lost, in a rate versus distortion sense, by basing the quantization and noiseless codes on the simple Laplacian model. These observations verify the suitability of pyramid codes [5]–[10] for DCT or subband image coding.

II. CODER STRUCTURE, MODELS, AND QUANTIZERS

This paper considers two intraframe image coder structures. A discrete cosine transform (DCT) coder [11] uses 8×8 pixel blocks and UTQ's designed for each coefficient. A subband coder [2], [12], [13] uses a seven-band octave decomposition obtained using Mallat's wavelet-based exact reconstruction filter bank [14] (a recomputed 31-tap version of Mallat's filter with coefficients listed in Appendix A). The lowest frequency subband (one sixteenth the size of the original image) is DCT encoded using 8×8 blocks, with UTQ's

Manuscript received December 6, 1992; revised October 19, 1993. This work was supported by the National Science Foundation, under Grants MIP-9116683, CDA-9121675, and NCR-8821764. The associate editor coordinating the review of this paper and approving it for publication was Prof. William A. Pearlman.

K. A. Birney is with the Hughes Aircraft Company, Aerospace and Defense Sector, Fullerton, CA 92634 USA.

T. R. Fischer is with the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752 USA.
IEEE Log Number 9407601.

again designed for each coefficient. The remaining subbands are uniform threshold quantized. Because the subband coder uses the DCT to encode the lowest frequency band, this coder will hereafter be referred to as a hybrid coder. The use of the DCT for encoding of the lowest frequency subband is motivated by the fact that most of the energy in natural images is contained in the lower frequency bands. The finer frequency resolution of the DCT is thus particularly well suited for coding the lowest frequency subband, in contrast with the higher frequency bands, where there is less variation in energy content with respect to frequency. Both coders use optimum rate allocation for the mean square error distortion measure, based on the coefficient models, as we will discuss later. Three different geometric models are studied for each coder. These are Laplacian, fixed generalized Gaussian, and adaptive generalized Gaussian models.

A. Laplacian Modeling

In order to define the Laplacian distribution approximating a given empirical distribution, the estimate of the mean absolute value of the data is computed. The corresponding distribution is then given as

$$p(x) = \frac{\lambda}{2} e^{-\lambda|x|} \tag{3}$$

where

$$\lambda = \frac{1}{E[|X|]} \tag{4}$$

assuming X is zero mean. This model is simple and mathematically tractable. The more complicated generalized Gaussian distribution will, however, always provide at least as good a fit to the empirical distribution to the extent that mean absolute value and variance can be accurately estimated.

B. Generalized Gaussian Modeling

The family of generalized Gaussian distributions is given by (1) and (2). A generalized Gaussian model can be chosen by matching the estimated mean absolute value and variance of the data set to those of a generalized Gaussian distribution. This is accomplished [14] by solving

$$\nu = F^{-1}\left(\frac{E[|X|]}{\sigma}\right) \tag{5}$$

where

$$F(\nu) = \frac{\Gamma(2/\nu)}{\sqrt{\Gamma(1/\nu)\Gamma(3/\nu)}} \tag{6}$$

and σ^2 is the estimated variance of the input signal. The values of ν and σ completely specify the generalized Gaussian distribution having the same first two moments as the empirical distribution. In practice, a look-up-table is used to invert $F(x)$ to solve for ν with reasonable precision. The software implementation studied accessed a 10 000-point table to cover ν values from 0.1 to 2.05.

C. Modeling of $D(R)$

In uniform threshold quantization, with a zero output level (a midtread quantization) and an arbitrarily large number of output levels, the step size completely specifies all threshold levels of the quantizer and parameterizes the rate distortion performance. The output entropy can be computed based on the probability of each output letter (obtained by integration of the source density). Similarly, the expected distortion is calculable after first computing the centroid of each decision region. The centroid is given by [15]

$$y_i = \frac{\int_{x_i}^{x_{i+1}} xp(x) dx}{\int_{x_i}^{x_{i+1}} p(x) dx} \tag{7}$$

where x_i and x_{i+1} are the lower and upper thresholds, respectively, of an arbitrary decision region. The expected distortion is then (for an L -level quantizer) [11]

$$D = \sum_{i=0}^{L-1} \int_{x_i}^{x_{i+1}} (x - y_i)^2 p(x) dx. \tag{8}$$

An analytical solution for the centroids and for the expected distortion exists for the Laplacian distribution, as closed-form antiderivatives can be found using standard techniques [15]. Numerical integration is required for solution of the generalized Gaussian case. The quantizer distortion-rate function can be modeled by [11]

$$D(\nu, R) = \gamma(\nu, R) \sigma^2 2^{-2R} \tag{9}$$

where $\gamma(\nu, R)$ depends on ν , which is the free parameter of the unit-variance generalized Gaussian probability density (see (1) and (2)).

D. Rate Allocation

In order to compare the impact of the choice of model on the performance of the coding system, software implementations were performed for the three modeling methods. For each respective model, the optimum encoding rates for the respective UTQ are computed as follows.

Notation: The rate allocated to a DCT coefficient will be referred to as R_j . Similarly, a DCT coefficient variance will be shown as σ_j^2 . The dc coefficient is assigned index zero. The order of assignment of the remaining coefficients is unimportant to the following development. In order to index DCT coefficients and subbands simultaneously, DCT coefficients will be assigned single indices and subbands double indices, that is, the k th subband of the j th level of a subband decomposition will have variance $\sigma_{j,k}^2$ and allocated rate $R_{j,k}$. The first level of decomposition ($j = 1$) corresponds to the three largest subbands that are one fourth the size of the original image. The second level is composed of the three subbands that are 1/16th of the original image size. Within a level of decomposition, the first subband ($k = 1$) captures vertical edges, the second captures horizontal edges, and the third diagonal edges and corners. The number of levels of decomposition is given by J , and N is the length of a side of a DCT block.

E. Adaptive Generalized Gaussian Model

This is the most complicated of the three approaches. It is "adaptive" in that it models the image signals to be encoded based on their estimated statistics. First, a generalized Gaussian parameter ν is computed for each subband and each non-dc DCT coefficient based on estimated mean absolute value and variance as described in (5) and (6). The dc coefficient is modeled as Gaussian ($\nu = 2.0$), as recommended by Reininger and Gibson [1]. Since dc coefficients represent only 1/64th of the DCT coefficients to be encoded, any mismatch that exists in the modeling of the dc coefficient does not have a significant impact on performance.

We confirmed that distortion is a decreasing function of output entropy for uniform threshold scalar quantization of generalized Gaussian sources by numerically computing rate-distortion curves for output entropies ranging from 0.01 to 5.0 b/pixel in increments of 0.01 b/pixel. This was done for values of ν from 0.3 to 1.3 in steps of 0.025 and for $\nu = 2.0$, which is a total of 42 different values of ν . The functions $\gamma(\nu, R)$ were then available by manipulating (9). A representative set of these functions is plotted in Fig. 1. The functions are modeled as constant for rates above 5.0 b/pixel. Since the distortion-rate functions are convex, the Kuhn-Tucker conditions [16] are satisfied. Thus, the rate allocation can be derived using Lagrangian techniques [11]. With the *a priori* assumption that $\nu_0 = 2.0$, the rate allocation equations are found to be

$$R_i = R + \frac{1}{2} \log_2 \frac{[2(\log_2 2)\gamma(\nu_i, R_i) - \partial\gamma(\nu_i, R_i)/\partial R_i]\sigma_i^2 2^{2j}}{A} \quad (10)$$

$$R_{j,k} = R + \frac{1}{2} \log_2 \frac{[2(\log_2 2)\gamma(\nu_{j,k}, R_{j,k}) - \partial\gamma(\nu_{j,k}, R_{j,k})/\partial R_{j,k}]\sigma_{j,k}^2 2^{2j}}{A} \quad (11)$$

where

$$A = \prod_{i=0}^{N^2-1} \left([2(\log_2 2)\gamma(\nu_i, R_i) - \frac{\partial\gamma(\nu_i, R_i)}{\partial R_i}]\sigma_i^2 2^{2j} \right)^{1/N^2 2^{2j}} \prod_{j=1}^J \prod_{k=1}^3 \left([2(\log_2 2)\gamma(\nu_{j,k}, R_{j,k}) - \frac{\partial\gamma(\nu_{j,k}, R_{j,k})}{\partial R_{j,k}}]\sigma_{j,k}^2 2^{2j} \right)^{1/2^{2j}} \quad (12)$$

In the event of a negative rate assignment, the most negative rate is set to zero, and the rate allocation is recalculated for the remaining signals.

Given the computed rate allocation, step sizes are selected such that the output of each UTC have entropy equivalent to its allocated rate. This is accomplished by forming a look-up table that gives the step size necessary to achieve a given output entropy for each source model. The quantizer output levels can then be computed as the centroids of the decision

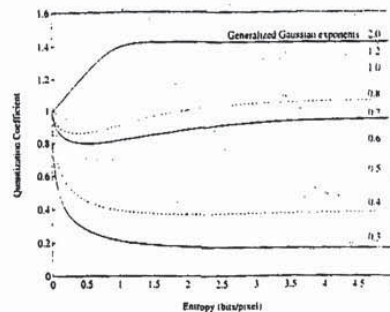


Fig. 1. Quantization Coefficients for various generalized Gaussian distributions.

regions. For the software implementation, step sizes have been calculated to achieve entropies from 0.01 to 5.0 b/pixel in steps of 0.01 b/pixel for each of the 42 values of ν mentioned previously. For allocations greater than 5.0 b/pixel, the step size can be approximated using the relation [17]

$$H = h(X) - \log_2 \Delta \quad (13)$$

where H is the entropy of the output of the quantizer, and Δ is the step size. Since the entropy of the generalized Gaussian source is [3]

$$h(X) = -\log_2 \left[\frac{\nu \eta(\nu, \sigma)}{2\Gamma(1/\nu)} \right] + \frac{1}{\nu \log_2 2} \quad (14)$$

the value of Δ to achieve a given H is found to be

$$\Delta = \frac{2^{1/\nu} \Gamma(1/\nu)^{3/2}}{\nu \Gamma(3/\nu)^{1/2}} 2^{-H} \quad (15)$$

The UTC step size is then available by multiplying Δ by the source standard deviation.

F. Fixed Generalized Gaussian Model

The fixed generalized Gaussian approach [2] assumes a constant distribution shape for each subband and non-dc DCT coefficient. Following Tanabe and Farvardin [2], we select $\nu = 0.7$ for the subbands and $\nu = 0.8$ for the non-dc 8×8 DCT coefficients of the lowest frequency subband for the hybrid coder case. In the 8×8 whole-image DCT case, we will also use $\nu = 0.8$ since the DCT size is the same. The dc coefficient is again modeled as Gaussian. This modeling approach thus uses only the tables for ν values of 0.7, 0.8, and 2.0 that were used for the adaptive generalized Gaussian case in order to select the step size yielding output entropy equal to the allocated rate for the given generalized Gaussian source.

The optimum rate allocation equations are available as a special case of (10)-(12) as

$$R_0 = R + \frac{1}{2} \log_2 \frac{[2(\log_2 2)\gamma(2.0, R_0) - \partial\gamma(2.0, R_0)/\partial R_0]\sigma_0^2 2^{2j}}{A} \quad (16)$$

$$R_i = R + \frac{1}{2} \log_2 \frac{[2(\log_e 2)\gamma(0.8, R_i) - \partial\gamma(0.8, R_i)/\partial R_i]\sigma_i^2 2^{2J}}{A} \quad (17)$$

$$R_{j,k} = R + \frac{1}{2} \log_2 \frac{[2(\log_e 2)\gamma(0.7, R_{j,k}) - \partial\gamma(0.7, R_{j,k})/\partial R_{j,k}]\sigma_{j,k}^2 2^{2J}}{A} \quad (18)$$

where

$$A = \left[2(\log_e 2)\gamma(2.0, R_0) - \frac{\partial\gamma(2.0, R_0)}{\partial R_0} \right] \sigma_0^2 2^{2J} \cdot \prod_{i=1}^{N^2-1} \left(\left[2(\log_e 2)\gamma(0.8, R_i) - \frac{\partial\gamma(0.8, R_i)}{\partial R_i} \right] \sigma_i^2 2^{2J} \right)^{1/N^2 2^{2J}} \prod_{j=1}^J \prod_{k=1}^3 \left([2(\log_e 2)\gamma(0.7, R_{j,k}) - \frac{\partial\gamma(0.7, R_{j,k})}{\partial R_{j,k}}] \sigma_{j,k}^2 2^{2J} \right)^{1/2^j} \quad (19)$$

C. Laplacian Model

Under the Laplacian assumption, all subband and non-dc DCT coefficient distributions are modeled as Laplacian. The dc coefficient is modeled as Gaussian for purposes of selecting the step size from the table. It is, however, grouped in with the rest of the coefficients (modeled as Laplacian) for the computation of rates. To further simplify the equations, it is assumed that $\gamma(R)$ in (9) is constant for all rates. The rate allocation becomes a highly simplified version of (10)-(12)

$$R_i = R + \frac{1}{2} \log_2 \frac{\sigma_i^2 2^{2J}}{A} \quad (20)$$

$$R_{j,k} = R + \frac{1}{2} \log_2 \frac{\sigma_{j,k}^2 2^{2J}}{A} \quad (21)$$

where

$$A = \prod_{i=0}^{N^2-1} (\sigma_i^2 2^{2J})^{1/N^2 2^{2J}} \prod_{j=1}^J \prod_{k=1}^3 (\sigma_{j,k}^2 2^{2J})^{1/2^j} \quad (22)$$

Only the tables for $\nu = 1.0$ and $\nu = 2.0$ are used by this modeling method. For rate allocations greater than 5.0 b/pixel, the step size is again computed using (13). For a unit variance Laplacian source, the equation reduces to

$$\Delta = \sqrt{2}e^{-H} \quad (23)$$

III. MODELING FOR ENTROPY CODING

One approach to entropy coding is to base a code on a model of an empirical distribution. This approach reduces the side information that must be transmitted in that only the parameters defining the model need be sent. Assuming an ideal entropy code is designed for the model distribution, excess bits will be used for data transmission due to imperfections in the model. The redundancy due to the use of the model can be quantified using the discrimination [4].

A. Quantification of Redundancy: The Discrimination

The discrimination is useful for quantifying the redundancy incurred in using a geometric distribution to model a source distribution for entropy coding. The discrimination, in units of bits per sample, is defined as [4]

$$L(q; \hat{q}) = \sum_j q_j \log_2 \frac{q_j}{\hat{q}_j} \quad (24)$$

where q represents the discrete distribution of the data, and \hat{q} is the vector of bin probabilities obtained by partitioning the geometric distribution using the same step size as was used for the uniform threshold quantization of the source. The discrimination is nonnegative [4] and provides a measure of the difference in bits between the entropy of the source and the average codeword length of the ideal noiseless code based on the model. The best a model of the source distribution can do is to introduce no additional redundancy to the entropy encoding.

B. Formation of Models

The model to be used for the noiseless encoding is consistent with that applied for purposes of quantization in the preceding section. For instance, the Laplacian coder uses a Laplacian model for both quantization and entropy coding. The empirical distributions are formed in each case by simply counting the number of occurrences of each output quantization level for the given source. A Laplacian or generalized Gaussian distribution defined by the values of ν or λ that were used in the modeling for quantization is then applied to model each source distribution. The geometric distributions are continuous and must be partitioned in order to form the appropriate probability vector. A vector of "bin" probabilities, which are suitable for comparison with the measured discrete distribution of the source, is thus obtained. Bin probabilities are calculated analytically for the Laplacian model and by using numerical integration for the generalized Gaussian models. The discrimination is then given by (24).

C. Side Information

All transform and subband variances and the DCT dc coefficient mean are assumed known, uniformly quantized with 16-bit accuracy, and transmitted as side information. An additional 6 b are required for each non-dc DCT coefficient in order to specify which of the allowable values of ν has been used to form the generalized Gaussian model. In the hybrid coding case, the non-dc coefficients are placed into four groups based on mean absolute value, and a single value of ν is assigned to each group. Two bits are required for each non-dc DCT coefficient to specify to which of the four groups each coefficient was assigned. The worst case occurs for DCT coding of 256 x 256 images, where the overhead is 0.0216 b/pixel. Since the side information is small, it is included only in the figures displaying PSNR performance (see Figs. 2, 3, 9, and 10).

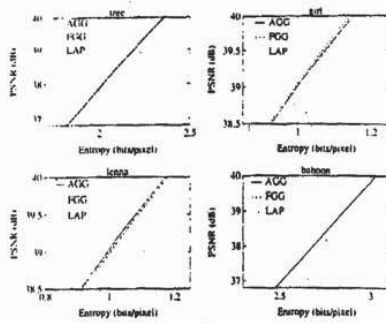


Fig. 2. PSNR performance of quantization for DCT coding.

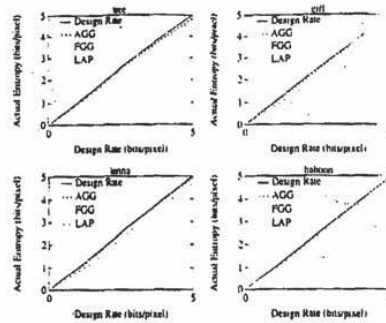


Fig. 4. Actual entropy versus design rate for DCT coding.

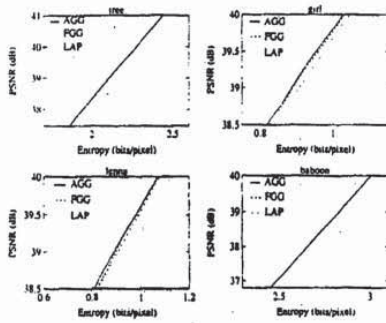


Fig. 3. PSNR performance of quantization for hybrid coding.

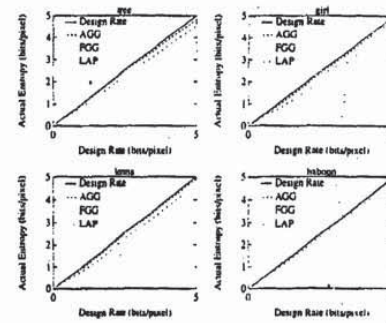


Fig. 5. Actual entropy versus design rate for hybrid coding.

IV. RESULTS

System performance was computed by running the algorithms on four different test images. The images were taken from the USC Image Data Base [18]. The *girl* (Image No. 5.1.1) and *tree* (5.1.6) are 256×256 , and *girl (Lenna)* (5.2.4) and *baboon* (5.2.3) are 512×512 images. The images were received in the 24-b red, green, blue color format. The standard linear transformation followed by 8-b linear quantization was applied to yield the luminance (black and white) images that were used in this study.

A. Quantization

The Laplacian, fixed generalized Gaussian, and adaptive generalized Gaussian techniques all provide nearly identical peak SNR performance at a given average output entropy over a wide range of design rates. A close look at Figs. 2 and 3 reveals that the largest performance gap between any two of the modeling techniques is on the order of 0.015 b/pixel for both DCT and hybrid coders. (The three techniques are abbreviated on the graphs as AGG, FGG, and LAP for adaptive generalized Gaussian, fixed generalized Gaussian,

and Laplacian, respectively.) One interesting case is the DCT coding of the *girl* image. Here, the fixed generalized Gaussian approach is observed to outperform the adaptive generalized Gaussian approach. This unusual case occurs because the adaptive approach requires an additional 0.0057 b/pixel of side information and because of the uncertainty in the estimates of the mean absolute values and variances of the DCT coefficients for the 256×256 image, which define the adaptive generalized Gaussian models.

Although the more complex adaptive generalized Gaussian model does not result in a significant gain in PSNR, it does provide a system that more accurately predicts the output entropy obtained by using particular step sizes to quantize the input signals. This results in an overall output entropy closer to the design rate, as observed in Figs. 4 and 5. See also Tables I and II for actual entropies at a design rate of 1 b/pixel.

All three methods had slightly more difficulty hitting the design rate for the hybrid coding case than for the whole-image DCT coder. This is again likely to be due to the uncertainty in parameter estimates that is brought on by the smaller number of data points on which to base estimates. Differences between the design rate and average output entropy are attributable

TABLE I
ACTUAL ENTROPY FOR DCT CODING AT A DESIGN RATE OF 1 b/pixel

Model	tree	girl	lenna	baboon
Laplacian	.9161	.8201	.7994	.5596
Fixed Generalized Gaussian	.9426	.8476	.8275	.9946
Adaptive Generalized Gaussian	.9795	.9527	.9471	.9955

TABLE II
ACTUAL ENTROPY FOR HYBRID CODING AT A DESIGN RATE OF 1 b/pixel

Model	tree	girl	lenna	baboon
Laplacian	.8535	.7655	.7873	.9486
Fixed Generalized Gaussian	.9032	.8126	.8527	1.0023
Adaptive Generalized Gaussian	.9604	.9281	.9307	.9999

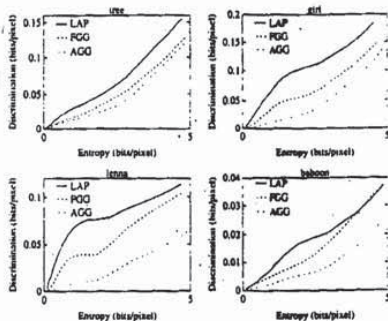


Fig. 6. Discrimination for DCT coding.

to mismatch between the respective models and empirical distributions. The adaptive generalized Gaussian approach clearly provides the best match to the design rate of the three modeling approaches. In applications where tolerance for variation from the design rate is low, the more complex adaptive generalized Gaussian modeling technique may prove useful.

B. Entropy Coding

The choice of model has a more significant impact on the performance of the noiseless encoding than on the performance of the quantization. However, as seen in Tables III through V, the advantage of the adaptive generalized Gaussian model at a design rate of 1 b/pixel reaches a maximum of only 0.0513 b/pixel. Examination of Figs. 6 and 7 shows that the performance gap between Laplacian and adaptive generalized Gaussian models never exceeds 0.08 b/pixel. The maximum gap occurs for hybrid coding of the girl image at a rate of about 2 b/pixel.

Fig. 8 displays discrimination observed in DCT coding of the lowest frequency subband for the hybrid coder. This case must be treated separately from the whole-image DCT as it occurs at a higher average entropy and because there are only 1/16th as many samples available on which to base parameter estimates. It is this added uncertainty in the estimation of

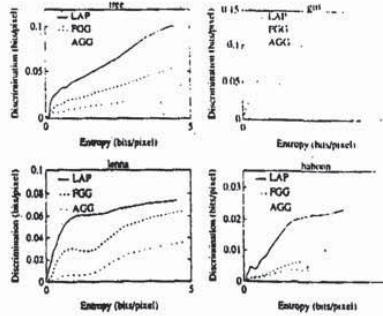


Fig. 7. Discrimination for hybrid coding.

TABLE III
DISCRIMINATION FOR DCT CODING AT A DESIGN RATE OF 1 b/pixel

Model	tree	girl	lenna	baboon
Laplacian	.0289	.0571	.0595	.0074
Fixed Generalized Gaussian	.0167	.0302	.0339	.0052
Adaptive Generalized Gaussian	.0135	.0113	.0082	.0025

TABLE IV
DISCRIMINATION FOR HYBRID CODING AT A DESIGN RATE OF 1 b/pixel

Model	tree	girl	lenna	baboon
Laplacian	.0513	.0813	.0659	.0119
Fixed Generalized Gaussian	.0440	.0591	.0413	.0087
Adaptive Generalized Gaussian	.0357	.0370	.0187	.0065

TABLE V
DISCRIMINATION FOR SUBBANDS ONLY FROM HYBRID CODING AT A DESIGN RATE OF 1 b/pixel

Model	tree	girl	lenna	baboon
Laplacian	.0327	.0568	.0500	.0090
Fixed Generalized Gaussian	.0181	.0316	.0278	.0028
Adaptive Generalized Gaussian	.0091	.0047	.0055	.0018

the parameters that is responsible for the behavior of the discriminations for the baboon image. The adaptive generalized Gaussian model incurs a greater discrimination than the fixed generalized Gaussian model because of the uncertainty in the parameter estimates that define the adaptive generalized Gaussian model.

It is also seen that the discriminations are not strictly increasing with increasing output entropy, although they are nearly so. This appears to be a consequence of the fact that the sample histograms are not strictly decreasing with increasing signal amplitude but are nearly so.

C. Overall System Performance

The overall quantization and entropy coding system peak SNR can now be examined with the discrimination values included. Through studying the effects of different distribution modeling methods on overall performance, the relative merits

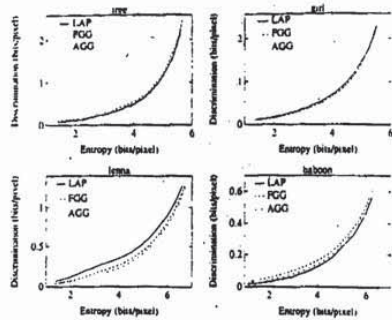


Fig. 8. Discrimination for subbands only from hybrid coding.

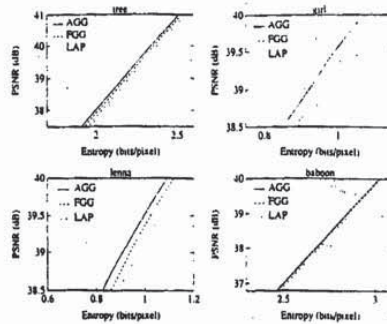


Fig. 10. Overall hybrid coding system PSNR performance.

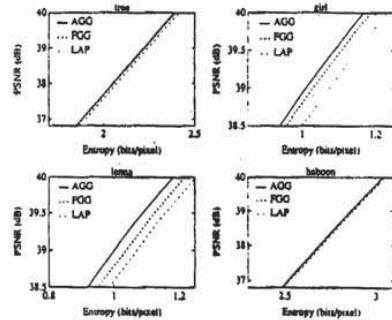


Fig. 9. Overall DCT coding system psnr performance.

of the modeling schemes can be examined. In all cases, ideal entropy coding of the dc DCT coefficient is assumed.

Comparing the different modeling techniques in Figs. 9 and 10, it is observed that generalized Gaussian modeling saves only on the order of 0.01 to 0.08 b/pixel over Laplacian modeling. Fixed generalized Gaussian modeling saves at most 0.03 b/pixel over Laplacian modeling. Nearly all of the savings in rate occurs in the noiseless encoding.

V. CONCLUSIONS

Modeling of source distributions is an effective way of simplifying quantizer and entropy code design for DCT and hybrid subband image coding. Adaptive generalized Gaussian modeling has a slight advantage over Laplacian modeling, ranging from about 0.01 to 0.08 b/pixel. Fixed generalized Gaussian outperforms Laplacian modeling by an even smaller margin (at most 0.03 b/pixel). Since Laplacian modeling has such a small cost relative to more complex approaches that apply the generalized Gaussian distribution, motivation is found for selecting pyramid codes for transform and subband image coding applications.

TABLE VI
FIR LOW PASS FILTER COEFFICIENTS (RECALCULATED FROM [14])

n	h(n)
0	0.5417355476
±1	0.3068293983
±2	-0.0354981886
±3	-0.0778081603
±4	0.0226844115
±5	0.0297465782
±6	-0.0121456978
±7	-0.0127156595
±8	0.0061412222
±9	0.0057990817
±10	-0.0030788380
±11	-0.0027455277
±12	0.0015440305
±13	0.0013306310
±14	-0.0005167136
±15	-0.0004363417

APPENDIX A

FINITE IMPULSE RESPONSE FILTER COEFFICIENTS

Table VI contains the finite impulse response filter coefficients.

APPENDIX B

IMAGE SOURCES

The images were taken from the USC Image Database [18]. The *girl* (Image No. 5.1.1) and *tree* (5.1.6) are 256 × 256, and *girl* (*Lenna*) (5.2.4) and *baboon* (5.2.3) are 512 × 512 images. The images were received in the 24-b red, green, blue color format. The standard linear transformation followed by the 8-b linear quantization was applied to yield the luminance (black and white) images used as input to the algorithms studied.

ACKNOWLEDGMENT.

The authors wish to thank S. Miller of Ampex Corporation for his assistance in providing test images and image display software.

REFERENCES

- [1] R. C. Reininger and J. D. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Transactions on Communications*, vol. COM-31, pp. 835-839, June 1983.
- [2] N. Tanabe and N. Farvardin, "Subband image coding using entropy-coded quantization over noisy channels," *Computer Science Technical Report Series*, University of Maryland, College Park, MD, Aug. 1989.
- [3] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Transactions on Information Theory*, vol. IT-30, pp. 485-497, May 1984.
- [4] R. E. Blahut, *Principles and Practice of Information Theory*. Reading, MA: Addison-Wesley, 1987.
- [5] T. R. Fischer, "A pyramid vector quantizer," *IEEE Transactions on Information Theory*, vol. IT-32, no. 4, pp. 568-583, July 1986.
- [6] H.-C. Tseng and T. R. Fischer, "Transform and hybrid transform/DPCM coding of images using pyramid vector quantization," *IEEE Transactions on Communications*, vol. COM-35, no. 1, pp. 79-86, Jan. 1987.
- [7] D. G. Jeong and J. D. Gibson, "Image coding with uniform and piecewise-uniform vector quantizers," submitted to *IEEE Transactions on Image Processing*.
- [8] M. Barlaud, P. Sole, M. Antonini, P. Mathieu, and T. Gaidon, "Pyramidal lattice vector quantization for multiscale image coding," submitted to *IEEE Transactions on Image Processing*.
- [9] T. R. Fischer and J. Pan, "Enumeration encoding and decoding algorithms for pyramid cubic lattice and trellis codes," submitted to *IEEE Transactions on Information Theory*.
- [10] D. G. Jeong and J. D. Gibson, "Uniform and piecewise uniform lattice vector quantization for memoryless Gaussian and Laplacian sources," *IEEE Transactions on Information Theory*, to appear.
- [11] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
- [12] J. W. Woods and S. D. O'Neil, "Subband coding of images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, no. 5, pp. 1278-1288, Oct. 1986.
- [13] J. C. Darragh, *Subband and Transform Coding of Images*. Ph.D. thesis, University of California, Los Angeles, 1989.
- [14] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674-693, July 1989.
- [15] M. D. Paez and T. H. Glisson, "Minimum mean-squared-error quantization in speech PCM and DPCM systems," *IEEE Transactions on Communications*, vol. COM-20, pp. 225-230, Apr. 1972.
- [16] B. D. Sivazlian and L. E. Stanfel, *Optimization Techniques in Operations Research*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
- [17] R. Wood, "On optimum quantization," *IEEE Transactions on Information Theory*, pp. 248-252, Mar. 1969.
- [18] A. Weber, "Image data base," *Report USC/PI 1070*, University of Southern California, Los Angeles, Mar. 1983.



Keith A. Birney was born in Greensburg, PA, in 1968. He received the B.S. and M.S. degrees in electrical engineering from Washington State University, Pullman, in 1989 and 1991, respectively. Since August 1991, he has been a Member of the Technical Staff at Hughes Aircraft Company, Fullerton, CA. His current research interests include image coding, digital signal processing, and digital communications.



Thomas R. Fischer (SM'88) was born in Fairbanks, Alaska, in 1953. He received the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Massachusetts, Amherst, and the Sc.B. degree magna cum laude from Brown University. From June 1975 until August 1976, he was a Staff Engineer at the Charles Stark Draper Laboratory, Cambridge, MA. From 1979 until 1988 he was with the Department of Electrical Engineering, Texas A&M University, first as Assistant Professor, then as Associate Professor. Since January 1989 he has been a Professor in the School of Electrical Engineering and Computer Science at Washington State University. His current research interests include data compression, digital communications, and digital signal processing. From 1989 to 1991 Professor Fischer was Associate Editor for Source Coding for the *IEEE Transactions on Information Theory*. He is a member and past Secretary of the Signal Processing and Communication Electronics Technical Committee of the IEEE Communications Society. In 1987 Professor Fischer received an outstanding teaching award from the College of Engineering at Texas A&M University.

Image Coding Using Wavelet Transforms and Entropy-Constrained Trellis-Coded Quantization

Parthasarathy Sriram, Member, IEEE, and Michael W. Marcellin, Senior Member

Abstract—The discrete wavelet transform has recently emerged as a powerful technique for decomposing images into various multi-resolution approximations. Multi-resolution decomposition schemes have proven to be very effective for high-quality, low bit-rate image coding. In this work, we investigate the use of entropy-constrained trellis-coded quantization (ECTCQ) for encoding the wavelet coefficients of both monochrome and color images. ECTCQ is known as an effective scheme for quantizing memoryless sources with low to moderate complexity. The ECTCQ approach to data compression has led to some of the most effective source codes found to date for memoryless sources.

Performance comparisons are made using the classical quadrature mirror filter bank of Johnston and nine-tap spline filters that were built from biorthogonal wavelet bases. We conclude that the encoded images obtained from the system employing nine-tap spline filters are marginally superior although at the expense of additional computational burden. Excellent peak-signal-to-noise ratios are obtained for encoding monochrome and color versions of the 512×512 "Lenna" image. Comparisons with other results from the literature reveal that the proposed wavelet coder is quite competitive.

I. INTRODUCTION

MULTI-FREQUENCY decomposition schemes are not new in the field of source coding. Subband coding was first introduced by Crochiere, Webber, and Flanagan [1] in 1976 for speech signals. The basic idea of any subband coding scheme is to decompose the input signal into a number of frequency bands (or subbands) using a bank of bandpass filters. Each subband is then decimated and encoded appropriately. At the receiver, the encoded subbands are interpolated and then passed through reconstruction filters to obtain the reconstructed signal. This approach, in general, demands the design of sophisticated bandpass filters to minimize the effects of aliasing.

Quadrature mirror filters (QMF) were introduced in [2] and allow alias free reconstruction of the signal in the absence of quantization errors. Vetterli [3] extended the application

Manuscript received August 1, 1993; revised June 7, 1994. This work was supported in part by the National Science Foundation under Grant Nos. NCR-8821764 and NCR-9258374, and by the Advanced Telecommunications Research Project at The University of Arizona. Portions of this material were presented at the SPIE conference on Visual Communications and Image Processing, Orlando, FL, Apr., 1992 and at the International Conference on Acoustics, Speech, and Signal Processing, Minneapolis, MN, Apr., 1993. The associate editor coordinating the review of this paper and approving it for publication was Prof. Nasser M. Nasrabadi.

P. Sriram is with Digital Communications Division, Rockwell International Corporation, Newport Beach, CA 92658 USA.

M. W. Marcellin is with the Department of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721 USA.
IEEE Log Number 9411136.

of QMF's to multi-dimensional signals. Both separable and nonseparable extensions were considered, but no coding results were presented. Subsequently, Woods and O'Neil [4] presented the first image coder using subband coding. The input image was split into 16 equal-sized subbands using circular convolution with 32-tap QMF's designed by Johnston [5]. They used DPCM to encode the image subbands.

Gharavi and Tabatabai [6] proposed another subband coding scheme in which the input image is split into seven unequal-sized subbands. The lowest frequency subband was encoded using DPCM while other subbands were encoded using memoryless quantizers. Their work was extended for color image coding as well. Since then, a variety of subband coders have emerged, capable of high-quality encoding with bit rates as low as 0.5 bits/pixel (e.g., [7]–[10]).

There are several advantages to multi-frequency decomposition schemes. Since quantization error variance can be separately controlled in each band by careful allocation of encoding rate, the overall reconstruction error spectrum can be controlled in such a manner that the reconstructed image is perceptually pleasing. Multi-resolution approximation schemes are also well suited for progressive image transmission [11].

A number of multi-resolution approximation schemes have emerged independently in different fields of engineering and science [12]. Recently, wavelet theory has been recognized as a unifying framework for these multi-resolution techniques [13]–[15]. Wavelets were originally introduced as a family of functions that were derived from translations and dilations of one basic function, referred to as the "mother" wavelet [16].

The basic idea of the discrete wavelet transform (DWT) is that of successive approximation, together with that of "added detail." At each stage, the input signal is decomposed into a coarse approximation signal (which can be considered a lowpass version of the input) and an "added detail" signal (which can be considered a highpass version). In this regard, the DWT decomposes the input signal into a set of frequency subbands [13].

Wavelet coders for images have been implemented both with scalar quantization [13] and vector quantization [17]. In this work, we investigate the use of trellis-coded quantization (TCQ) with the DWT for encoding both monochrome and color images. TCQ was recently introduced as an effective scheme for quantizing memoryless sources with low to moderate complexity [18]. The motivation for TCQ comes from Ungerboeck's formulation of trellis-coded modulation (TCM) [19]. As in TCM, an expanded codebook is partitioned into subsets, and these subsets are used to label the branches of

an appropriate trellis. For a given data sequence, the Viterbi algorithm [20] is then used to find the minimum mean squared error (MSE) path through the trellis.

In order to apply wavelet decompositions to images, we use a separable 2-D DWT in which emphasis is given to the horizontal and vertical directions. For the monochrome case, we evaluate the performance of our wavelet coder for seven-band and 16-band decompositions. In each case, the lowest frequency sub-image (LFS) is encoded using a 2-D discrete cosine transform (DCT) encoder (with a block size of 4×4) while the other sub-images are encoded using TCQ for memoryless data. An integer programming algorithm [21] is employed to allocate the available bit-rate optimally among the subbands. A small amount of side information, consisting of the sample mean of the "DC" coefficients and the sample standard deviation of all sub-images to be encoded, is transmitted. The procedural flow for color images is similar, except for the conversion of the RGB planes into NTSC transmission primaries (Y, I, and Q).

Our preliminary results with TCQ of image subbands were reported in [22] and proved to be quite competitive with other encoding techniques from the literature. In that work, 33-tap filters based on Mallat's wavelet [13] were employed for the subband decomposition. The wavelet image coder reported in [17] used short-length filters based on biorthogonal wavelets [23]. In this work, we use the nine-tap spline filters suggested in [23] and [17] for the wavelet decomposition. We also compare, from both a MSE calculation and a subjective judgment, the performance of nine-tap spline filters and the traditional QMF's of Johnston [5].

II. DISCRETE WAVELET TRANSFORM

A DWT utilizes two functions: the mother wavelet ψ and a scaling function ϕ . The scaling function ϕ can be chosen in such a manner that the translations of (dilated versions of) ϕ form a basis for a vector space, say \tilde{V}_m . Letting m vary results in a sequence of successive approximation spaces, i.e.

$$\dots \subset \tilde{V}_2 \subset \tilde{V}_1 \subset \tilde{V}_0 \subset \tilde{V}_{-1} \subset \tilde{V}_{-2} \dots \quad (1)$$

\tilde{V}_m is said to have a resolution of 2^{-m} . Translations of ψ span a vector space \tilde{W}_m that is the complement of \tilde{V}_m in \tilde{V}_{m-1} .

The approximation of an arbitrary input function f at a resolution 2^{-m} (say $A_m(f)$) is given by the projection of f onto the vector space \tilde{V}_m . The information lost when going from an approximation of f with resolution 2^{-m} to the coarser approximation $A_{m+1}(f)$ with resolution $2^{-(m+1)}$ is referred to as the error, or "detail" signal $D_{m+1}(f)$, and can be obtained by the projection of f onto \tilde{W}_{m+1} . The detail signal $D_{m+1}(f)$ is typically a highpass version of f while $A_{m+1}(f)$ is a lowpass version.

Let \tilde{h} and \tilde{g} be the impulse responses for decomposition (or analysis) lowpass and highpass filters, respectively. Given the approximation of f at a resolution of 2^{-m} (i.e., $A_m(f)$), $A_{m+1}(f)$ and $D_{m+1}(f)$ can be computed by filtering $A_m(f)$ with \tilde{h} and \tilde{g} and then keeping every other sample of the output. This algorithm is illustrated by the block diagram

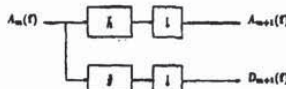


Fig. 1. Block diagram of a wavelet decomposition.

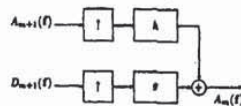


Fig. 2. Block diagram of a wavelet reconstruction.

shown in Fig. 1. Approximations at lower resolutions are obtained by repeated application of this algorithm.

Let ψ and ϕ be the wavelet and the scaling function necessary for reconstruction. Given $A_{m+1}(f)$ and $D_{m+1}(f)$, $A_m(f)$ can be perfectly reconstructed by interpolating $A_{m+1}(f)$ and $D_{m+1}(f)$ by a factor of two and filtering the resulting signals with \tilde{h} and \tilde{g} , respectively. The block diagram shown in Fig. 2 illustrates this algorithm. Figs. 1 and 2 reveal that discrete wavelet transforms are essentially subband decomposition systems. Hence, the terms "subband decomposition" and "wavelet decomposition" are used interchangeably throughout this paper.

It is advantageous to have wavelet bases that are orthonormal. In that case, the sub-images are orthogonal. In addition, for image processing applications, one would prefer analysis and synthesis filters to have linear phase. Unfortunately, there exist no nontrivial, finite-length, orthogonal linear-phase filters with the perfect reconstruction property [17]. In practice, this difficulty is overcome by either dropping the necessity for perfect reconstruction or by using biorthogonal bases [23], [17].

Biorthogonal bases for wavelets were recently introduced, independently, by Cohen, Daubechies, and Feauveau [23] and by Vetterli and Herley [24]. In [23], it was shown that it is possible to construct bases that yield finite-length, linear-phase filters with the perfect reconstruction property by relaxing the orthonormality requirement. Decomposition and reconstruction filters for the resulting "biorthogonal" bases are related by

$$H(\omega)\tilde{H}(\omega) = \cos(\omega/2)^{2l} \left[\sum_{p=0}^{l-1} \binom{l-1+p}{p} \sin(\omega/2)^{2p} + \sin(\omega/2)^{2l} R(\omega) \right] \quad (2)$$

and

$$g(n) = (-1)^n \tilde{h}(-n+1), \quad \tilde{g}(n) = (-1)^n h(-n+1)$$

where H and \tilde{H} are the Fourier transforms of h and \tilde{h} , $R(\omega)$ is an odd polynomial in $\cos(\omega)$, $2l = k + \tilde{k}$, and the functions ψ and $\tilde{\psi}$ are $(k-1)$ and $(\tilde{k}-1)$ continuously differentiable, respectively. $H(\omega)$ and $\tilde{H}(\omega)$ can be chosen

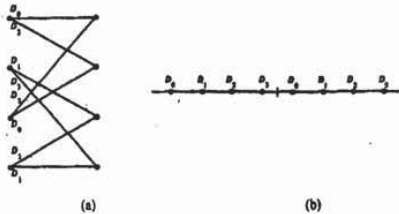


Fig. 3. A four-state trellis with subset labeling (a); codebook and partition for two-bit per sample TCQ (b).

in several different ways [23]. We have chosen to use the spline variant family of filters suggested in [17] and [23] with $k = \bar{k} = 4$. For this selection, \hat{h} and \bar{h} are nine-tap and seven-tap filters, respectively. For convenience, these filters are referred to as nine-tap spline filters for the remainder of this paper.

III. TRELLIS-CODED QUANTIZATION

A. Fixed-Rate TCQ

The motivation for TCQ comes from Ungerboeck's formulation of trellis-coded modulation (TCM) [19]. In the simplest case, for encoding a memoryless source using TCQ at a rate of R bits/sample, a scalar codebook having 2^{R+1} elements is partitioned into four subsets (each containing 2^{R-1} codewords). These subsets are then used to label the branches of a suitably chosen trellis. An example of a four-state trellis with corresponding codebook and partition (for $R = 2$ bits/sample) is shown in Fig. 3. The justification for these particular choices is provided in [18].

For a given sequence of data, the Viterbi algorithm [20] is used to find the sequence of codewords (as allowed by the trellis structure) that minimizes the MSE between the data and the selected codeword sequence. One obvious method to encode the resulting sequence of TCQ codewords into a bit sequence is to allocate one bit/sample for specifying a path through the trellis (equivalently, a sequence of subsets) while using the remaining $R-1$ bits/sample to specify a codeword from the subset chosen at each point in time. A more detailed description of TCQ and an explanation for why it achieves excellent performance as an algorithm for data compression can be found in [18].

B. Entropy-Constrained TCQ

Entropy-constrained TCQ (ECTCQ) was introduced in [25]. In that work, near optimal performance (in a rate-distortion theory sense) for encoding memoryless sources was achieved at encoding rates greater than about 1.5 bits/sample. As explained in the previous subsection, one bit was used to specify a path through the trellis with the remaining available rate being allocated to specifying an element from each subset. The entropy of the codeword elements in each subset were then computed as an estimate to this remaining portion of the rate. Obviously, encoding rates for scalar codebooks are

greater than one bit/sample for this scheme. Lower rates were achieved using vector codebooks.

In an attempt to achieve encoding rates between 0.0 and 1.0 bits/sample with scalar codebooks, a different approach was followed in [26] that enabled ECTCQ to achieve near optimal performance for encoding memoryless sources at all nonnegative encoding rates. This new approach makes use of the fact that in any given state, the next codeword must be chosen from either $S_0 = D_0 \cup D_2$ or $S_1 = D_1 \cup D_3$. For example, when in the top state (of the trellis in Fig. 3), the next codeword must be chosen from S_0 . S_0 and S_1 are called *supersets*. Rather than using one bit/sample to specify a path through the trellis with the remaining rate allocated to selecting elements from the chosen subsets, all the available rate can be used to specify an element from a superset. Since the subsets are disjoint, specifying an element from a superset uniquely determines which subset the codeword comes from, and therefore, the next trellis state. One variable-length code is provided for each of the two supersets, and an estimate of the rate required to encode data under this scheme is given by the conditional entropy of the codebook given the superset:

$$H(\hat{X}|S) = - \sum_{i=0}^1 \sum_{s \in S_i} P(\hat{x}|S_i) P(S_i) \log_2 P(\hat{x}|S_i). \quad (3)$$

IV. IMAGE CODING APPLICATION

A block diagram illustrating the procedural flow for a monochrome TCQ wavelet coder is shown in Fig. 4. The input image is decomposed into a series of sub-images using a 2-D DWT. Since images are spatially limited, the filtering and decimation result in an expansion of data. To circumvent this problem, we have used a generalization of the symmetric extension technique described in [8] and [27] that allows the amount of data to be reduced to its original size while introducing no distortion.

A similar system using Johnston's QMF's [5] was studied in [10]. In that work, both DPCM and DCT were used to encode the lowest frequency sub-image (LFS). The DCT based system was found to be superior. We have followed their approach and use a 2-D DCT with a block size of 4×4 for encoding the LFS. All "like" DCT coefficients of the LFS are collected into sequences to be encoded using ECTCQ. Each of the high-frequency sub-images (HFS) is also treated as a sequence to be encoded (with no further processing) using ECTCQ. In each case, four-state ECTCQ systems are used.

A small amount of side information, consisting of the sample mean of the "DC" transform coefficient and the sample standard deviation of all sub-images and DCT coefficients, is transmitted. All DCT coefficients and sub-images are normalized by subtracting their mean (all data except the DC transform coefficient are assumed to be zero-mean) and then dividing by their respective standard deviations. The "normalized" transform coefficients and sub-images are then encoded using ECTCQ at rates determined by the optimum rate-allocation scheme described in a subsequent section. At the receiving end, the resulting bit sequence and normalization parameters (side information) are used to reconstruct the

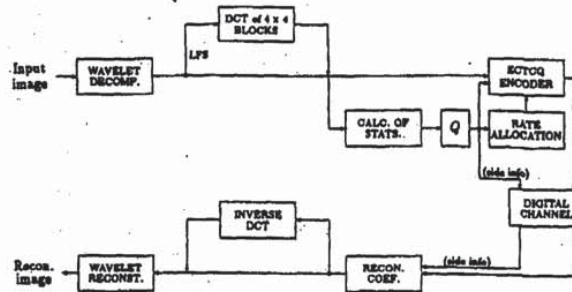


Fig. 4. Block diagram of a monochrome TCQ wavelet coder.

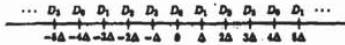


Fig. 5. Uniform codebook and partition for an ECTCQ system.

quantized coefficients. The inverse DCT is performed to obtain the reconstructed LFS before the final wavelet reconstruction stage.

A. Codebook Design

In [25] and [26], it was shown that for encoding memoryless sources with smooth densities (using ECTCQ), near-optimal performance (in a rate-distortion theory sense) can be obtained by employing the codebook design algorithm and encoding rule from [28]. This algorithm attempts to minimize the MSE of an encoding (but subject to an entropy constraint) by minimizing the cost function

$$J = E[\rho(x, \hat{x})] + \lambda E[l(\hat{x})] \quad (4)$$

where x is the data, \hat{x} is the encoded version of x , $\rho(x, \hat{x})$ is the cost (usually MSE) of representing x by \hat{x} , λ is a Lagrange multiplier, and $l(\hat{x})$ is the number of bits used by the variable-length code to represent \hat{x} .

In [26], it was found that for rates greater than 2.5 bits/sample, the optimized codebooks do not provide a significant improvement in MSE over uniform codebooks (as shown in Fig. 5). Thus, in our simulations with images, optimized codebooks were used for encoding rates less than or equal to three bits/sample while uniform codebooks were used for all other encoding rates.

A collection of 30 images (different from the "Lenna" image) were used as training data for the optimization algorithm. Three sets of codebooks—one for the DC coefficient, one for the other DCT coefficients and the third for all sub-images other than the LFS—were used. For each set, codebooks were designed for integer multiples of 0.1 bits/sample. The number of bits required to represent a particular codeword \hat{x} was computed as

$$l(\hat{x}|S_i) = \frac{1}{\log_2} [-10 \log_2 P(\hat{x}|S_i)] \quad (5)$$

where $P(\hat{x}|S_i)$ (the probability of using \hat{x} given that the superset S_i is used) is estimated from the training data.

Let $\hat{P}(\hat{x}|S_i)$ be the relative frequency of \hat{x} (given superset S_i) for encoding test data. Similarly, let $\hat{P}(S_i)$ be the relative frequency of S_i for encoding test data. It is then easy to show that

$$\bar{l} = \sum_i \sum_j \hat{P}(\hat{x}|S_i) \hat{P}(S_i) l(\hat{x}|S_i)$$

is an upper bound to the encoding rate (for the test data) required by two Huffman codes (designed from the training data for S_0 and S_1 , respectively) each operating on blocks of codewords with length ten.

As pointed out before, the TCQ decoder always knows whether the next codeword should come from S_0 or S_1 . Thus, the S_0 codewords can be collected into a sequence, Huffman coded, and transmitted, followed by the same procedure for the S_1 codewords. At the decoder, the two sequences can be Huffman decoded. The TCQ decoder can then sequentially decode the data by drawing codewords from the S_0 and S_1 sequences as appropriate.

B. Rate Allocation

The basic intention of any rate allocation scheme is to appropriately allocate the bits to be used for encoding among the data sequences to be encoded so as to optimize the performance according to some objective cost function. In this work, we use a bit allocation algorithm in which the distortion-rate performances of different quantizers are used [21]. This algorithm produces an optimal or very nearly optimal allocation, while allowing the set of admissible bit allocation values to be constrained to a finite set of nonnegative numbers.

Specifically, the overall MSE incurred by our coding scheme is given by

$$E = \sum_{i=1}^K \alpha_i w_i E_i(r_i) \quad (6)$$

where $E_i(r_i)$ is used to denote the distortion-rate performance for encoding the i th data sequence at r_i bits/sample, K is the

number of data sequences¹, and α_i is a weighting coefficient to account for the variability in the size of the sequences. Also, since the biorthogonal synthesis filters h and g do not have the same energy, the quantization noise in various subbands will not be equally weighted in the image reconstruction. The scaling factor w_i is introduced to offset this disparity. A detailed treatment of the procedure to find these weighting coefficients for a given set of filters can be found in [29].

In practice, the rate allocation vector $B = (r_1, r_2, \dots, r_K)$ is chosen so as to minimize E subject to the constraint that

$$\sum_{i=1}^K \alpha_i r_i \leq R \text{ bits/pixel.} \quad (7)$$

In [21], it is shown that the solution $B^* = (r_1^*, r_2^*, \dots, r_K^*)$ to the unconstrained problem

$$\min_B \left\{ \sum_{i=1}^K \alpha_i w_i E_i(r_i) + \lambda \sum_{i=1}^K \alpha_i r_i \right\} \quad (8)$$

minimizes E subject to $\sum_{i=1}^K \alpha_i r_i \leq \sum_{i=1}^K \alpha_i r_i^*$. Thus, to find a solution to the constrained problem of (6) and (7), it suffices to find λ such that the solution to (8) yields $\sum_{i=1}^K \alpha_i r_i^* \leq R$. A detailed treatment of an algorithm to find the proper λ can be found in [21].

For a given λ , the solution to the unconstrained problem is obtained by minimizing each term of the sum in (8) separately. If $V_k = \{p_k, \dots, q_k\}$ is the set of allowable rates for the k th quantizer, then r_k^* solves

$$\min_{r_k \in V_k} \{ \alpha_k w_k E_k(r_k) + \lambda \alpha_k r_k \}. \quad (9)$$

C. Side Information

The side information consists of the sample mean of the DC transform coefficient and the sample standard deviation of all data sequences to be encoded. A 16-bit uniform quantizer was used to quantize each of these parameters resulting in $16(K+1)$ bits/image of side information. In addition, the initial trellis state for each data sequence needs to be transmitted to the receiver [30]. For a four-state trellis, this requires $2K$ bits/image. Hence, the overall side information amounts to $(18K+16)$ bits/image that corresponds to 0.002 bits/pixel for a 16-band decomposition of a monochrome 512×512 image.

V. RESULTS AND CONCLUSIONS

A. Monochrome Image Coding

Coding simulations were performed for the luminance component of the 512×512 "Lenna" image. The performance of our image coder is reported by tabulating the peak signal-to-noise ratio (PSNR), which is defined as

$$\text{PSNR} = 10 \log_{10} \left(\frac{255^2}{\text{MSE}} \right) \text{ dB.} \quad (10)$$

¹ K is the number of subbands minus one, plus the number of DCT coefficient sequences. For example, K equals 22 and 31 for seven- and 16-band decompositions, respectively.

TABLE I
PERFORMANCE OF JOHNSTON'S FILTERS FOR ENCODING THE "LENA" IMAGE

8-tap		16-tap		24-tap		32-tap	
Obtained Rate	PSNR	Obtained Rate	PSNR	Obtained Rate	PSNR	Obtained Rate	PSNR
0.48	33.34	0.48	38.28	0.47	38.44	0.47	38.70
0.27	31.68	0.27	33.04	0.27	33.87	0.27	34.01

For color images, the MSE is computed as an average over all three image planes (RGB).

Subband coders that have been proposed in the literature have used both seven-band (pyramidal) and 16-band (tree-structured) decompositions. Westerink, Biemond, and Boeckee in [31] compared different decomposition schemes in a fixed-rate coding system using QMF's and reported that the best objective performance is obtained when the image is split into 16 equally sized subbands. It is not mentioned in [31], however, if there is any improvement in the subjective quality of the encoded images.

We investigated the performance of our wavelet coder using a seven-band (7B) and a 16-band (16B) decomposition. To obtain the same PSNR value, the 7B system required an encoding rate approximately 15% higher than that of the 16B system. Interestingly, even at equal PSNR, the images obtained from the 16B system are sharper and have less high frequency background noise than those from the 7B system. One possible explanation for this occurrence is the fact that PSNR is not very sensitive to noise in the HFS because of their low energy content. On the other hand, these sub-images contain significant edge information and if not quantized efficiently, introduce ringing and high-frequency fuzziness. High frequency sub-images are encoded more efficiently by the 16B system than by the 7B system. Thus, for a given PSNR, the 16-band decomposition results in an improvement in both the encoding rate and the quality of the reconstructed imagery. All simulations from this point forward, assume the use of a 16-band decomposition.

Recently, Westerink, Biemond, and Boeckee [32] analyzed the use of QMF's of different lengths on aliasing distortions in a subband image coding application using Johnston's filters and scalar quantization. Comparisons were made at encoding rates of 0.8 and 0.6 bits/pixel. They concluded that from both an MSE calculation and a subjective judgment aliasing errors can be neglected for filter lengths of 12 taps or more. They also reported that for encoding rates of at least 0.8 bits/pixel, the effect of aliasing distortions in image subbands is negligible. The encoder in [32] uses Lloyd-Max quantizers to encode the subbands and does not exploit any correlation in the LFS.

Our subband coding system is significantly different than the system in [32]. As a result, their conclusions may not be valid for our system. We simulated our system using Johnston's 8, 16, 24, and 32-tap filters. The results for encoding the 512×512 "Lenna" image for "desired" rates of 0.5 and 0.25 bits/pixel are shown in Table I. The obtained rates are different from the desired rates specified in the rate allocation procedure because of the entropy-constrained design of the TCQ systems.

It is obvious from Table I that the PSNR performance of the 32- and 24-tap systems are superior to the 16- and eight-tap systems. Recall from our discussion in the previous section

TABLE II
WAVELET CODING RESULTS FOR ENCODING THE MONOCHROME "LENA" IMAGE

Desired rate	Obtained rate	PSNR
1.00	0.95	36.33
0.50	0.48	36.61
0.25	0.27	32.77

that a QMF bank using Johnston's filters is not a perfectly reconstructing filter bank even in the absence of quantization distortions. This distortion (QMF distortion) increases as the filter length becomes smaller. QMF distortion is typically not perceptible in the reconstructed image. Hence, the fact that the PSNR of the reconstructed image from the 32-tap system is higher does not necessarily translate to higher quality reconstructed imagery.

Subjective tests revealed that, at both encoding rates, images obtained from the eight-tap system were the best. At approximately 0.5 bits/pixel, the encoded images from all systems were of extremely high-quality. However, the systems based on the longer filter kernels create "ringing" near edges. At 0.27 bits/pixel, these Gibbs phenomena type errors affected the quality of the encoded images considerably when the long filters were employed. In smooth regions of the image, all four encoded images seem to have approximately the same amount of perceptible distortion.

We also implemented our subband coder using the nine-tap spline filters. Simulation results for encoding the 512×512 "Lenna" image are shown in Table II for "desired" rates of 1.0, 0.5, and 0.25 bits/pixel. Comparing results in Tables I and II, it is obvious that the performance of the spline filters is comparable to the performance of the 24-tap system while it is uniformly better than the performance of 16- and eight-tap systems.

A subjective evaluation of the encoded images revealed that the encoded images obtained from the system employing nine-tap spline filters are marginally better than those obtained from the eight-tap system². This improvement in the subjective quality could be attributed to the regularity and differentiability of scaling functions associated with the spline filters [17]. More details regarding the importance of regularity and differentiability of scaling functions in an image coding application can be found in [17]. Due to their superior performance, all results from this point forward assume use of the spline filters for the wavelet decomposition.

In comparison with other results from the literature, we find that our wavelet coder is quite competitive. The simulation results from Table II (plus two additional points described below) are shown in Fig. 6 along with other results from the literature. At an encoding rate of 0.48 bits/pixel, our PSNR value of 36.61 dB is higher than those of the entropy-constrained scalar quantization based subband coder of Tanabe and Farvardin [10] (35.32 dB at 0.45 bits/pixel), the ECTCQ based three-component model image coder in [34] (36.53 dB at

²Note that this improvement is obtained at the expense of additional computational complexity. For a two-band decomposition of one-dimensional data, generating one output sample from the lowpass and highpass analysis filters requires eight multiplies and eight adds when using eight-tap Johnston's filters (by using the polyphase decomposition technique suggested in [33]) while spline filters require nine multiplies and 14 adds.

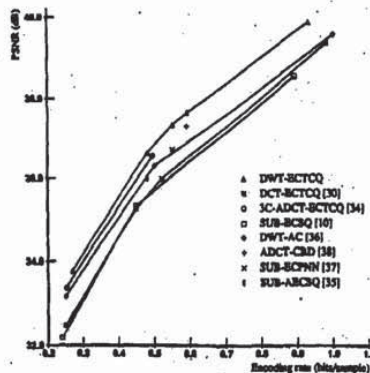


Fig. 6. Comparisons for encoding the 512×512 "Lenna" image.

0.495 bits/pixel), the ECTCQ based transform coder proposed in [30] (35.97 dB at 0.52 bits/pixel), the adaptive entropy coded subband coder of Kim and Modestino [35] (35.98 dB at 0.48 bits/pixel), and the embedded wavelet image coder of Shapiro [36] (35.97 dB at 0.50 bits/pixel). In [37] and [38], PSNR's of 36.70 dB and 37.3 dB were reported at 0.55 and 0.59 bits/pixel, respectively. When our system is simulated at these rates, PSNR values of 37.33 dB and 37.63 dB are obtained, for improvements of 0.63 and 0.33 dB, respectively.

The subjective quality of the encoded images is excellent. The encoded image at an average rate of 0.93 bits/pixel is almost indistinguishable from the original image and the encoded image at 0.48 bits/pixel is extremely good with very little high-frequency background noise or smoothing. The original image is shown in Fig. 7 while the image encoded at 0.48 bits/pixel is shown in Fig. 8. There are no visible artifacts even when viewed on a high-resolution monitor. The encoded image at 0.27 bits/pixel is quite natural looking but has some perceptual distortion.

B. Perceptual Weighting

One of the most important objectives of an image coding system is to encode images in such a manner that coding distortions are not perceptible. To compress an image such that a human observer cannot perceive coding distortions is not an easy task. One must understand the psychophysics of the human visual system (HVS) to achieve this. It is known that the sensitivity of the human eye in perceiving distortion is different for different spatial frequencies [39]. We can make use of this information in our subband coder by perceptually weighting each subband according to the sensitivity of the human eye to the energy in that subband. We follow the ideas of Perkins and Lookabaugh [40] for calculating these weighting coefficients and modify the previously discussed rate allocation algorithm to incorporate these coefficients. That is, instead of appropriately allocating the available rate among the data sequences to be encoded so as to minimize MSE given



Fig. 7. Monochrome "Lenna" image (512 × 512).



Fig. 8. Encoded image at 0.48 bits/pixel (PSNR = 36.61 dB).

by (6), the weighted MSE (WMSE), defined as

$$\text{WMSE} = \sum_{i=1}^K \alpha_i w_i p_i E_i(r_i) \quad (11)$$

is minimized subject to the constraint of (7) where p_i is a perceptual weighting coefficient for the i th band.

On the monitor used to view images, a 512 × 512 image is of size 128 mm × 128 mm. We set the viewing distance to be four times the size of the image (i.e., $d = 512$ mm). We investigated the choice of weighting coefficients associated

with the DCT bands by computing them in three different ways. They are:

- 1) All DCT bands were given the same weighting coefficient, the one corresponding to the center frequency of the LFS.
- 2) Weighting coefficients for the DCT bands were found using the fact that a 4 × 4 block is of size 1 mm × 1 mm. For a viewing distance of d , the observer's eye subtends an angle α given by

$$\tan\left(\frac{\alpha}{2}\right) = \frac{1/2}{d} = \frac{1}{1024}$$

Hence, $\alpha = 2 \tan^{-1}(1/1024) = 0.1119^\circ$. The function $\cos[\pi k(2m+1)/2N]$ will complete k cycles in N samples (and hence in α°). As a result, the frequency in cycles/degree is given by $c_k = k/\alpha = 8.9366k$. Similarly, the value of c corresponding to the 2-D DCT coefficient $X_{\text{dct}}(k, l)$ is given by $c = \sqrt{(8.9366k)^2 + (8.9366l)^2} = 8.9366\sqrt{k^2 + l^2}$.

- 3) Frequencies obtained from method (2) were scaled in such a manner that the value of c corresponding to the highest-frequency DCT band is the same as the one corresponding to the highest-frequency DFT coefficient ($X(127, 127)$) in the LFS. Weighting coefficients corresponding to the scaled c 's were calculated as before.

The DC band was always given a weight of one because it determines the average intensity of the block. Subjective tests revealed that the encoded images obtained from using weighting coefficients from method (3) appear the best. We obtained PSNR values of 36.37 and 33.47 dB at encoding rates of 0.53 and 0.28 bits/pixel, respectively, from such a system. Comparing these results with those in Table II, it is evident that perceptual weighting results in a small drop in PSNR values for approximately the same encoding rate. At approximately 0.5 bits/pixel, it was very difficult to identify any improvement in perceptual quality due to weighting. This is not very surprising as the encoded images are of extremely high quality. However, at an encoding rate of approximately 0.25 bits/pixel, the encoded image from the perceptually weighted system looks better. This improvement in subjective quality increases as the encoding rate is decreased further.

A primary effect of perceptual weighting is to emphasize low frequencies with respect to high frequencies. An image like "Lenna" has little high-frequency content. As a result, the effect of perceptual weighting might be exaggerated. We encoded the "baboon" image and an aerial image of an urban area, both of which have significant high-frequency content. As before, the effect of weighting was very difficult to perceive at 0.5 bits/pixel. However at approximately 0.25 bits/pixel, encoded images from the perceptually weighted system were significantly better.

C. Color Image Coding

It is well known that the three color planes (red, green, and blue) are highly correlated. To exploit this redundancy, it is a common practice to transform these planes to the NTSC transmission primaries specified as Y, I, and Q. This

TABLE III
PERFORMANCE RESULTS FOR ENCODING THE COLOR "LENA" IMAGE

System		Design bit rates		
		0.25	0.5	1.0
ECTCQ based wavelet coder	Obtained bit rate	0.24	0.47	1.13
	PSNR	35.44	35.72	36.20
ECTCQ based transform coder [30]	Obtained bit rate	0.25	0.49	1.0
	PSNR	35.38	35.51	34.85

transform has the added advantage of being compatible with monochrome television (Y component). As in the monochrome case, each component is decomposed into 16 equal-sized subbands. The LFS for each component is encoded using a 2-D DCT encoder with a block size of 4×4 .

It is well known that the human eye is less sensitive to degradation in the chrominance components than to the degradation in the luminance component. As a result, color image coders concentrate on encoding the luminance component more efficiently than the chrominance components. The subband coder proposed in [6] discards all HFS associated with the I and Q components while all subbands of the Y component are encoded. At the decoder, the chrominance components are restored to their original size by interpolation.

We investigated the significance of high-frequency bands associated with the chrominance components both perceptually and in an MSE sense by implementing our wavelet coder in the following manner:

- 1) All high-frequency sub-images of I and Q components were encoded (48B);
- 2) All high-frequency sub-images of I and Q components were discarded (18B).

The performance of the two systems is approximately equal (both objectively and subjectively) at low encoding rates (≈ 0.25 bits/pixel). This is as expected since even for the 48B system, the HFS receive zero encoding rates from the rate allocation algorithm because of their very low variance. At high rates (≈ 1.0 bit/pixel), discarding the HFS associated with the chrominance components causes a significant drop in PSNR and effects the subjective quality of the encoded images in an interesting way. Without side-by-side comparison with the original, the encoded image from the 18B system looks extremely good. However, careful comparison with the original reveals that colors have a lighter, or "washed out" appearance. The 48B system does not suffer from this effect. Hence, in the simulations discussed below, the high-frequency subbands associated with the chrominance components were not discarded.

Simulation results are presented in Table III for encoding the color version of the 512×512 "Lena" image at three different encoding rates. For comparison, the performance of the ECTCQ-based transform coder proposed in [30] is also shown. It is evident from this table that the PSNR performance of our wavelet coder is superior to the system in [30] at all encoding rates. The comparison with the results from [30] is not completely fair because the encoder in [30] subsamples the chrominance components by a factor of two in each direction before quantization.

The subjective quality of the encoded images at all three rates is extremely good. In particular, the encoded image at 1.13 bits/pixel is indistinguishable from the original. The encoded image at 0.47 bits/pixel is extremely sharp and devoid of any annoying artifacts. Fuzziness and high-frequency background noise are totally absent even at an average encoding rate of 0.24 bits/pixel.

REFERENCES

- [1] R. E. Crochiere, S. A. Webber, and J. L. Flanagan, "Digital coding of speech in subbands," *Bell Syst. Tech. J.*, vol. 55, pp. 1069-1085, Oct. 1976.
- [2] A. Croisier, D. Esteban, and C. Galand, "Perfect channel splitting by use of interpolation/decimation/tree decomposition techniques," in *Conf. Proc., 1976 IEEE Int. Conf. Inform. Sci. Syst.*, Patras, Greece, May 1976.
- [3] M. Vetterli, "Multi-dimensional subband coding: Some theory and algorithms," *Signal Processing*, vol. 6, pp. 97-112, Apr. 1984.
- [4] J. W. Woods and S. D. O'Neill, "Subband coding of images," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1278-1286, Oct. 1986.
- [5] J. D. Johnston, "A filter family designed for use in quadrature mirror filter banks," in *Conf. Proc., 1980 Int. Conf. Acoust., Speech, Signal Processing*, Denver, CO, Apr. 1980.
- [6] H. Charari and A. Tabatabai, "Sub-band coding of monochrome and color images," *IEEE Trans. Circuit Syst.*, vol. 35, pp. 207-217, Feb. 1988.
- [7] P. H. Weserink, D. E. Boekae, J. Blomond, and J. W. Woods, "Subband coding of images using vector quantization," *IEEE Trans. Commun.*, vol. 36, pp. 713-719, June 1988.
- [8] M. J. T. Smith and S. L. Eddins, "Analysis/synthesis techniques for subband image coding," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1446-1456, Aug. 1990.
- [9] S. Nanda and W. A. Pearlman, "True coding of image subbands," *IEEE Trans. Image Processing*, vol. IP-1, pp. 133-147, Apr. 1992.
- [10] N. Tambe and N. Farvardin, "Subband image coding using entropy-coded quantization over noisy channels," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 926-943, June 1992.
- [11] E. Simoncelli and E. H. Adelson, "Nonseparable extensions of quadrature mirror filters to multiple dimensions," *Proc. IEEE*, vol. 78, pp. 652-664, Apr. 1990.
- [12] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Mag.*, vol. 8, pp. 14-38, Oct. 1991.
- [13] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 11, pp. 674-693, Jul. 1989.
- [14] ———, "Multifrequency channel decomposition of images and wavelet models," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 2091-2110, Dec. 1989.
- [15] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, pp. 909-996, Nov. 1988.
- [16] A. Grossmann and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM J. Math.*, vol. 15, pp. 723-736, July 1984.
- [17] M. Antonini, M. Beraud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205-220, Apr. 1992.
- [18] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of markovian and Gauss-Markov sources," *IEEE Trans. Commun.*, vol. 38, pp. E2-93, Jan. 1990.
- [19] G. Ungerböck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, vol. 28, pp. 55-67, Jan. 1982.
- [20] G. D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. PROC-61, pp. 268-278, Mar. 1973.
- [21] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1445-1453, Sept. 1988.
- [22] P. Srinam and M. W. Marcellin, "Wavelet coding of images using trellis coded quantization," in *Conf. Proc., 1992 SPIE Conf. Visual Commun. Image Processing*, Orlando, FL, Apr. 1992.
- [23] A. Cohen, I. Daubechies, and J. C. Feauveau, "Biorthogonal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. 41, pp. 485-540, June 1992.
- [24] M. Vetterli and C. Herley, "Wavelets and filter banks: Relationships and new results," in *Conf. Proc., 1990 Int. Conf. Acoust., Speech, Signal Processing*, Albuquerque, NM, Apr. 1990.

- [25] T. R. Fischer and M. Wang, "Entropy-constrained trellis-coded quantization," *IEEE Trans. Inform. Theory*, vol. 38, pp. 415-426, Mar. 1992.
- [26] M. W. Marcellin, "On entropy-constrained trellis coded quantization," *IEEE Trans. Commun.*, vol. 42, pp. 14-16, Jan. 1994.
- [27] S. A. Martucci, "Signal estimation and noncausal filtering for subband coding of images," in *Conf. Proc., 1991 SPIE Conf. Visual Commun. Image Processing*, Boston, MA, Nov. 1991.
- [28] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Entropy-constrained vector quantization," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 31-42, Jan. 1989.
- [29] J. W. Woods and T. Naveen, "A filter based bit allocation scheme for subband compression of HDTV," *IEEE Trans. Image Processing*, vol. 1, pp. 436-440, Jul. 1992.
- [30] M. W. Marcellin, P. Sriram, and K.-L. Tong, "Transform coding of monochrome and color images using trellis coded quantization," *IEEE Trans. Circuits Syst. (Video Technol.)*, vol. 3, pp. 270-276, Aug. 1993.
- [31] P. H. Westerink, J. Biemond, and D. E. Boelke, "Evaluation of image subband coding schemes," in *Conf. Proc., European Signal Processing Conf., Grenoble, France, Sept. 1988*.
- [32] ———, "Scalar quantization error analysis for image subband coding using QMF's," *IEEE Trans. Signal Processing*, vol. 40, pp. 421-427, Feb. 1992.
- [33] P. P. Vaidyanathan, "Multirate digital filters, filter banks, polyphase networks, and applications: A tutorial," *Proc. IEEE*, vol. 78, pp. 56-93, Jan. 1990.
- [34] N. Farvardin, X. Ran, and C.-C. Lee, "Adaptive DCT coding of images using entropy-constrained trellis coded quantization," in *Conf. Proc., 1993 Int. Conf. Acoust. Speech, Signal Processing*, Minneapolis, MN, Apr. 1993.
- [35] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing*, vol. 1, pp. 31-48, Jan. 1992.
- [36] J. M. Shapiro, "An embedded wavelet hierarchical image coder," in *Conf. Proc., 1992 Int. Conf. Acoust. Speech, Signal Processing*, San Francisco, CA, Mar. 1992.
- [37] D. P. de Garrido, W. A. Pearlman, and W. A. Finamore, "Vector quantization of image pyramids with the ECPNN algorithm," in *Conf. Proc., 1991 SPIE Conf. Visual Commun. Image Processing*, Boston, MA, Nov. 1991.
- [38] W. A. Pearlman, P. Jakasdar, and M. M. Leung, "Adaptive transform tree coding of images," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 902-912, June 1992.
- [39] A. N. Netravali and B. G. Haskell, *Digital Pictures, Representation and Compression*. New York: Plenum Pr., 1988.
- [40] M. G. Perkins and T. Lookabaugh, "A psychophysically justified bit allocation algorithm for subband image coding," in *Conf. Proc., 1989 Int. Conf. Acoust., Speech, Signal Processing*, May 1989.



Parthasarathy Sriram (S'85-M'93) was born in Trichy, India, on July 7, 1967. He received the B.Eng. degree in electronics and communications engineering from Regional Engineering College, Trichy, India, and the M.S. and Ph.D. degrees in electrical engineering from the University of Arizona, Tucson, in 1990 and 1993, respectively. Since 1993 he has been with the Digital Communications Division, Rockwell International, Newport Beach, CA. His research interests include data compression, digital communication, and digital signal processing.



Michael W. Marcellin (S'81-M'82-SM'93) was born in Bishop, CA, on July 1, 1959. He received the M.S. and Ph.D. degrees in electrical engineering from Texas A&M University in 1985 and 1987, respectively. He graduated summa cum laude with the B.S. degree in Electrical Engineering from San Diego State University in 1983, where he was named the most outstanding student in the College of Engineering. While at Texas A&M, he received a fellowship for graduate study from the General Electric Company.

Dr. Marcellin joined the Department of Electrical and Computer Engineering at the University of Arizona in 1988, where he is currently an associate professor. His research interests include digital communication and data storage systems, data compression, and signal processing.

Dr. Marcellin is a member of Tau Beta Pi, Eta Kappa Nu, and Phi Kappa Phi. He is a 1992 recipient of the National Science Foundation Young Investigator Award, and a corecipient of the 1993 IEEE Signal Processing Society Senior Award.

AN OVERVIEW OF WAVELET BASED MULTIRESOLUTION ANALYSES*

BJÖRN JAWERTH^{†‡} AND WIM SWELDENS^{†§}

Abstract. In this paper we present an overview of wavelet based multiresolution analyses. First, we briefly discuss the continuous wavelet transform in its simplest form. Then, we give the definition of a multiresolution analysis and show how wavelets fit into it. We take a closer look at orthogonal, biorthogonal and semiorthogonal wavelets. The fast wavelet transform, wavelets on an interval, multidimensional wavelets and wavelet packets are discussed. Several examples of wavelet families are introduced and compared. Finally, the essentials of two major applications are outlined: data compression and compression of linear operators.

Key words. wavelet, multiresolution analysis, compression

AMS subject classifications. 42-02, 42C10

Contents.

1. Introduction
2. Notation
3. A short history of wavelets
4. The continuous wavelet transform
5. Multiresolution analysis
6. Orthogonal wavelets
7. Biorthogonal wavelets
8. Wavelets and polynomials
9. The fast wavelet transform
10. Examples of wavelets
11. Wavelets on an interval
12. Wavelet packets
13. Multidimensional wavelets
14. Applications

1. Introduction. Wavelets have generated a tremendous interest in both theoretical and applied areas, especially over the past few years. The number of researchers, already large, continues to grow, so progress is being made at a rapid pace. In fact, advancements in the area are occurring at such a rate that the very meaning of "wavelet analysis" keeps changing to incorporate new ideas.

In a rapidly developing field, overview papers are particularly useful, and several good ones concerning wavelets are already available, such as [60, 83, 115, 122, 123, 125]. Of these, [122] contains a brief introduction to multiresolution analysis, [60] describes wavelets from an approximation theory point of view, [83] discusses continuous and discrete wavelets, [125] focuses on the construction of wavelets, [115] looks at wavelets from a signal processing point of view and [123] compares wavelets with Fourier techniques.

* Manuscript received by the editors ??, accepted for publication ??

[†] Department of Mathematics, University of South Carolina, Columbia SC 29208

[‡] This author is partially supported by DARPA Grant AFOSR F49620-93-1-0083 and ONR Grant N00014-90-J-1343.

[§] Departement Computerwetenschappen, Katholieke Universiteit Leuven, Celestijnenlaan 200 A, B 3001 Leuven, Belgium. This author is Research Assistant of the National Fund of Scientific Research Belgium, and is partially supported by ONR Grant N00014-90-J-1343 and the Experimental Program to Stimulate Competitive Research (EPSCOR NSF Grant EHR-9108772).

Our paper differs from these in that it contains some more recent developments and that it focuses on the "multiresolution analysis" aspect of wavelets. The emphasis on multiresolution analysis allows us to look at a number of different constructions of wavelets such as orthogonal, semiorthogonal, and biorthogonal wavelets. By examining these constructions in a unified setting, we are ideally positioned to make comparisons between them. The recent developments contain wavelets on an interval, multidimensional wavelets, and wavelet packets.

We have selected an expository style and a level of rigor that we hope will present the ideas without obscuring them in too much detail. Instead of giving exact, detailed statements in a "theorem-structured" way, we have opted for a more informal style. References are given throughout, pointing to more details when needed.

For example, this paper occasionally contains statements of the form "A is (essentially) equivalent with B". The interpretation that we have in mind is that, for "all practical purposes", A is equivalent to B. Strictly speaking, the equivalence may only hold under some extra technical conditions. Good examples are when a formula is guaranteed to be true only "almost everywhere" or in a "weak sense".

The style we have chosen is motivated by the intended audience: people with a more theoretical interest as well as those working in various applied areas. For a reader in the first category this paper might provide some of the theory, and point to some of the right references for further study. A reader in the second category could use the paper to make comparisons, and find connections to related material.

The paper is organized as follows. After a brief sketch of the history of wavelets we introduce the "continuous wavelet transform." The discussion of the continuous wavelet transform is mainly included for historical purposes and for comparison with the multiresolution analysis wavelets. Next, we give the definitions of "multiresolution analysis" and "scaling function" (Section 5), derive some basic properties and illustrate these with some examples. In this section we also give the basic definition of "wavelet." Wavelets are then studied in more detail in the next sections. Section 6 discusses orthogonal wavelets, while Section 7 treats biorthogonal wavelets, a generalization of the orthogonal ones, and semiorthogonal wavelets, a compromise between the previous two. In the following section we study the connection between wavelets and polynomials, and show how this relates to the approximation properties of wavelet expansions. In Section 9 we show how a "fast wavelet transform" can be derived from the multiresolution analysis properties. In the appendix, the reader can find a pseudocode implementation of this algorithm. At this point, i.e. after the study of the basic properties of multiresolution analysis, we are ready to single out some desirable properties of wavelets. This is done in Section 10. We also give several examples of wavelet families, such as Daubechies' and spline wavelets, and compare their properties. The next three sections focus on more recent developments such as wavelets on an interval, wavelet packets and multidimensional wavelets. These sections can be read independently. Finally, in the last section (Section 14) we consider the basic ideas associated with two important applications: data compression and analysis of linear operators.

It goes without saying (almost) that this short overview is still highly incomplete. It is unfortunate that we were unable to cover many other important and interesting developments in the area, some of which are more significant than the ones we have included. For example, we hardly mention the significant volume of work done in the direction of approximation theory, and the efforts in the field of fractal functions and the more applied areas are left out almost entirely. We apologize to the people whose

results we were unable to discuss due to the constraints imposed by the overview format.

Finally, let us point out that, although wavelets are a relatively recent phenomenon, there are a number of useful sources of information about them. First of all, there are three new journals with an emphasis on wavelets: Applied Computational Harmonic Analysis, Journal of Fourier Analysis, and Advances in Computational Mathematics. Secondly, several journals have had special issues on wavelets, such as Constructive Approximation, IEEE Transactions on Signal Processing, IEEE Transactions on Information Theory, International Journal of Optical Computing, Journal of Mathematical Imaging and Vision, and Optical Engineering. Also, an electronic information service exists on the Internet, the Wavelet Digest, with the address wavelet@math.sc Carolina.edu. Last but not least, several books on the subject exist, monographs as well as edited volumes. The list includes [13, 20, 21, 43, 49, 74, 92, 96, 106, 108, 116, 119].

2. Notation. Most of the notation will be presented as we go along. The space of square integrable functions, $L^2(\mathbb{R})$, is defined as the space of Lebesgue measurable functions for which

$$\|f\|^2 = \int_{-\infty}^{+\infty} |f(x)|^2 dx < \infty.$$

The inner product of two functions $f, g \in L^2(\mathbb{R})$ is given by

$$\langle f, g \rangle = \int_{-\infty}^{+\infty} f(x) \overline{g(x)} dx,$$

and the Fourier transform of a function $f \in L^2(\mathbb{R})$ is defined as

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(x) e^{-i\omega x} dx.$$

The Poisson summation formula is used in the following two forms,

$$\sum_l f(x-l) = \sum_k \hat{f}(2k\pi) e^{i2k\pi x},$$

and

$$\sum_l \langle f, g(\cdot - l) \rangle e^{-i\omega l} = \sum_k \hat{f}(\omega + k2\pi) \overline{\hat{g}(\omega + k2\pi)}.$$

If no bounds are indicated under a summation sign, $\in \mathbb{Z}$ is understood.

A countable set $\{f_n\}$ of a Hilbert space is a *Riesz basis* if every element f of the space can be written uniquely as $f = \sum_n c_n f_n$, and positive constants A and B exist such that

$$A \|f\|^2 \leq \sum_n |c_n|^2 \leq B \|f\|^2.$$

3. A short history of wavelets. The history of wavelets could be the topic of a separate paper. Let us give a short, subjective account.

Wavelet theory involves representing general functions in terms of simpler, fixed building blocks at different scales and positions. This has been found to be a useful

approach in several different areas. For example, we have subband filtering techniques, quadrature mirror filters, pyramid schemes, etc., in signal and image processing, while in mathematical physics similar ideas are studied as part of the theory of Coherent States. Wavelet theory represents a useful synthesis of these different approaches.

In abstract mathematics, it has been known for quite some time that techniques based on Fourier series and Fourier transforms are not quite adequate for many problems and so-called *Littlewood-Paley techniques* often are effective substitutes. These techniques were initially developed in the 30's to understand, among other things, summability properties of Fourier series and boundary behavior of analytic functions. In the 50's and 60's, these developed into powerful tools for studying other things, such as solutions of partial differential equations and integral equations. It was realized that they fit into *Calderón-Zygmund theory*, an area of harmonic analysis that is still very heavily researched.

One of the standard approaches, not only in Calderón-Zygmund theory, but in analysis in general, is to break up a complicated phenomenon into many simple pieces and study each of the pieces separately. In the 70's, sums of simple functions, called atomic decompositions [35], were widely used, especially in Hardy space theory. One method used to establish that a general function f has such a decomposition, is to start with the "Calderón formula": for a function f , one has that

$$f(x) = \int_0^{+\infty} \int_{-\infty}^{+\infty} (\psi_t * f)(y) \tilde{\psi}_t(x-y) dy \frac{dt}{t}.$$

The $*$ denotes convolution. Here $\psi_t(x) = t^{-1}\psi(x/t)$, and $\tilde{\psi}_t(x)$ is defined similarly, for appropriate fixed functions ψ and $\tilde{\psi}$. As we shall see below, this representation is an example of a continuous wavelet transform. In mathematical physics the Aslaksen-Klauder construction of the $(ax+b)$ -coherent states can be seen as another independent derivation of the Calderón formula [7, 91].

In the early 80's, Strömberg discovered the first orthogonal wavelets [126]. This was done in the context of trying to further understand Hardy spaces, as well as other spaces used to measure the size and smoothness of functions. A discrete version of the Calderón formula had also been used for similar purposes in [86] and long before this there were results by Haar [81], Franklin [70], Ciesielski [26], Peetre [112], and others.

Independent from these developments in harmonic analysis, Alex Grossmann, Jean Morlet, and their coworkers studied the wavelet transform in its continuous form [78, 79, 80]. The theory of "frames" [51] provided a suitable general framework for these investigations.

In the early to mid 80's, several groups, perhaps most notably the one associated with Yves Meyer and his collaborators, independently realized, with some excitement, that tools from Calderón-Zygmund theory, in particular the Littlewood-Paley representations, had discrete analogs and could give a unified view of many of the results in harmonic analysis. Also, one started to understand that these techniques could be effective substitutes for Fourier series in numerical applications. (The first named author of this paper came to this understanding through the joint work with Mike Frazier [71, 72, 73].) As the emphasis shifted more towards the representations themselves, and the building blocks involved, the name of the theory also shifted. Alex Grossmann and Jean Morlet suggested the word "wavelet" for the building blocks, and what earlier had been referred to as Littlewood-Paley theory, now started to be called wavelet theory.

Pierre-Gilles Lemarié and Yves Meyer [97], independent of Strömberg, constructed new orthogonal wavelet expansions. With the notion of multiresolution analysis, introduced by Stéphane Mallat and Yves Meyer, a systematic framework for understanding these orthogonal expansions was developed [103, 104, 105]. It also provided the connection with quadrature mirror filtering. Soon, Ingrid Daubechies [47] gave a construction of wavelets, non-zero only on a finite interval and with arbitrarily high, but fixed, regularity. This takes us up to a fairly recent time in the history of wavelet theory. Several people have made substantial contributions to the field over the past few years. Some of their work and the appropriate references will be discussed in the body of the paper.

4. The continuous wavelet transform. Since we are going to be brief, let us start by pointing out that more detailed treatments of the continuous wavelet transform can be found in [20, 77, 78, 83]. As mentioned above, a wavelet expansion uses translations and dilations of one fixed function, the wavelet $\psi \in L^2(\mathbf{R})$. In the case of the continuous wavelet transform, the translation and dilation parameters vary continuously. In other words, the transform makes use of the functions

$$\psi_{a,b}(x) = \frac{1}{\sqrt{|a|}} \psi\left(\frac{x-b}{a}\right) \quad \text{with } a, b \in \mathbf{R}, a \neq 0.$$

These functions are scaled so that their $L^2(\mathbf{R})$ norms are independent of a . The continuous wavelet transform of a function $f \in L^2(\mathbf{R})$ is now defined by

$$(1) \quad \mathcal{W}(a, b) = \langle f, \psi_{a,b} \rangle.$$

Using Parseval's identity, we can also write this as

$$(2) \quad 2\pi \mathcal{W}(a, b) = \langle \hat{f}, \hat{\psi}_{a,b} \rangle,$$

where

$$\hat{\psi}_{a,b}(\omega) = \frac{a}{\sqrt{|a|}} e^{-i\omega b} \hat{\psi}(a\omega).$$

We assume now that the wavelet ψ and its Fourier transform $\hat{\psi}$ are functions with finite centers \bar{x} and $\bar{\omega}$ and finite radii Δ_x and Δ_ω . These quantities are defined by

$$\bar{x} = \frac{1}{\|\psi\|^2} \int_{-\infty}^{+\infty} x |\psi(x)|^2 dx,$$

$$\Delta_x^2 = \frac{1}{\|\psi\|^2} \int_{-\infty}^{+\infty} (x - \bar{x})^2 |\psi(x)|^2 dx,$$

and similarly for $\bar{\omega}$ and Δ_ω . The variable x usually represents either time or space; we shall settle for the first and refer to x as time. From (1) and (2), we see that the continuous wavelet transform at (a, b) picks up information about f , mostly from the time interval $[b + a\bar{x} - a\Delta_x, b + a\bar{x} + a\Delta_x]$ and from the frequency interval $[(\bar{\omega} - \Delta_\omega)/a, (\bar{\omega} + \Delta_\omega)/a]$. These two intervals determine a *time-frequency window*. Its width, height and position are governed by a and b . Its area is constant and given by $4\Delta_x \Delta_\omega$. The Heisenberg uncertainty principle says that this area has to be greater than 2. These time-frequency windows are also called *Heisenberg bozes*.

Suppose that the wavelet ψ satisfies the *admissibility condition*

$$C_\psi = \int_{-\infty}^{+\infty} \frac{|\widehat{\psi}(\omega)|^2}{\omega} d\omega < \infty.$$

Then, the continuous wavelet transform $\mathcal{W}(a, b)$ is invertible on its range, and an inverse transform is given by the relation

$$(3) \quad f(x) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathcal{W}(a, b) \psi_{a,b}(x) \frac{da db}{a^2}.$$

From the admissibility condition, we see that $\widehat{\psi}(0)$ has to be 0, and, in particular, ψ has to oscillate. This, together with the decay property, has given ψ the name *wavelet* or "small wave" (French: *ondelette*). This shows that the frequency localization of the wavelets is much better than pointed out above. In most cases $\bar{\omega}$ is zero and the frequency localization is really in a band $[-\omega_2/a, -\omega_1/a] \cup [\omega_1/a, \omega_2/a]$, because $\widehat{\psi}(0)$ vanishes. This can help to understand why a reconstruction formula of type (3) is possible.

The transform is often represented graphically and plotted as two two-dimensional images with color or grey-scale value corresponding to the modulus and phase of $\mathcal{W}(a, b)$. This representation has been used extensively in areas such as geophysics.

In applications, it is of interest to find inverse transforms that do not make use of \mathcal{W} over the whole range of a and b . Transforms exist that only use positive values of a or even only discrete values for a . Furthermore, using the theory of frames it is possible to study the case where only discrete values for a and b are used, see [83] for an excellent overview. The most common choice is to use a dyadic grid, i.e. to let $a = 2^{-j}$ and $b/a = l$ with $j, l \in \mathbb{Z}$ [48, 73]. In general, the fewer values of a and b one wants to use, the more restrictive the condition on the wavelet becomes. The continuous wavelet transform allows us to use a very general wavelet. At the other extreme, we shall see that much more restrictive conditions hold for a wavelet used in multiresolution analysis. This allows us, on the other hand, to prove powerful results such as the construction of orthogonal bases.

The transform that only uses the dyadic values of a and b was originally called the *discrete wavelet transform*. At this moment, however, this term is ambiguous, since it is also used to denote the transform from the sequence of scaling function coefficients of a function to its wavelet coefficients (see Section 9).

The case when a , and b belong to more irregular sets have also been covered. Such irregular sampling results can be found in [14, 39, 68, 67].

The continuous wavelet transform is also used in *singularity detection* and characterization [71, 100]. A typical result in this direction is that if a function f is Hölder (Lipschitz) continuous of order $0 < \alpha < 1$, so that $|f(x+h) - f(x)| = \mathcal{O}(h^\alpha)$, then the continuous wavelet transform has an asymptotic behavior like

$$\mathcal{W}(a, b) = \mathcal{O}(a^{\alpha+1/2}) \quad \text{for } a \rightarrow 0.$$

The converse is true as well. The advantage of this characterization compared to the Fourier transform is that it does not only provide information about the kind of singularity, but also about its location in time. There is a corresponding characterization of Hölder (Lipschitz) continuous functions of higher order $\alpha \geq 1$; the wavelet must then have a number of vanishing moments greater than α , i.e.

$$\int_{-\infty}^{+\infty} \psi(x) x^p dx = 0 \quad \text{for } 0 \leq p \leq \alpha \quad \text{and } p \in \mathbb{Z}.$$

We note that the number of vanishing wavelet moments limits the order of smoothness that can be characterized.

Example. A classical example of a wavelet is the *Mexican hat* function,

$$\psi(x) = (1 - 2x^2)e^{-x^2}.$$

Being the second derivative of a Gaussian, it has two vanishing moments.

5. Multiresolution analysis.

5.1. The scaling function and the subspaces V_j . There are at least two ways to introduce wavelets: one is through the continuous wavelet transform as in the previous section, and another is through multiresolution analysis. Here we start by defining multiresolution analysis, and then point out some of the connections with the continuous wavelet transform.

A *multiresolution analysis* of $L^2(\mathbb{R})$ is defined as a sequence of closed subspaces V_j of $L^2(\mathbb{R})$, $j \in \mathbb{Z}$, with the following properties [47, 103]:

1. $V_j \subset V_{j+1}$,
2. $v(x) \in V_j \Leftrightarrow v(2x) \in V_{j+1}$,
3. $v(x) \in V_0 \Leftrightarrow v(x+1) \in V_0$,
4. $\bigcup_{j=-\infty}^{+\infty} V_j$ is dense in $L^2(\mathbb{R})$ and $\bigcap_{j=-\infty}^{+\infty} V_j = \{0\}$,

5. A *scaling function* $\varphi \in V_0$, with a non-vanishing integral, exists such that the collection $\{\varphi(x-l) \mid l \in \mathbb{Z}\}$ is a Riesz basis of V_0 .

The references [122, 123] contain an introduction to the concept of multiresolution analysis.

Let us make a couple of simple observations concerning this definition. Since $\varphi \in V_0 \subset V_1$, a sequence $(h_k) \in \ell^2(\mathbb{Z})$ exists such that the scaling function satisfies

$$(4) \quad \varphi(x) = 2 \sum_k h_k \varphi(2x - k).$$

This functional equation goes by several different names: the *refinement equation*, the *dilation equation* or the *two-scale difference equation*. We shall use the first.

It is immediate that the collection of functions $\{\varphi_{j,l} \mid l \in \mathbb{Z}\}$, with $\varphi_{j,l}(x) = \sqrt{2^j} \varphi(2^j x - l)$, is a Riesz basis of V_j .

By integrating both sides of (4), and dividing by the (non-vanishing) integral of φ , we see that

$$(5) \quad \sum_k h_k = 1.$$

If the scaling function belongs to L^1 , it is, under very general conditions, uniquely defined by its refinement equation and the normalization [52],

$$\int_{-\infty}^{+\infty} \varphi(x) dx = 1.$$

In many cases, no explicit expression for φ is available. However, there are fast algorithms that use the refinement equation to evaluate the scaling function φ at dyadic points ($x = 2^{-j}k$, $j, k \in \mathbb{Z}$) [15, 18, 47, 52, 53, 122]. In many applications, we never need the scaling function itself; instead we may often work directly with the h_k .

The spaces V_j will be used to approximate general functions. This will be done by defining appropriate projections onto these spaces. Since the union of all the V_j is dense in $L^2(\mathbf{R})$, we are guaranteed that any given function can be approximated arbitrarily close by such projections.

To be able to use the collection $\{\varphi(x-l) \mid l \in \mathbf{Z}\}$ to approximate even the simplest functions (such as constants), it is natural to assume that the scaling function and its integer translates form a *partition of unity*, or, in other words,

$$(6) \quad \forall x \in \mathbf{R} : \sum_k \varphi(x-k) = 1.$$

Note that by Poisson's summation formula, the partition of unity is (essentially) equivalent with

$$(7) \quad \widehat{\varphi}(2\pi k) = \delta_k \quad \text{for } k \in \mathbf{Z}.$$

By (4), the Fourier transform of the scaling function must satisfy

$$(8) \quad \widehat{\varphi}(\omega) = H(\omega/2) \widehat{\varphi}(\omega/2),$$

where H is a 2π -periodic function defined by

$$(9) \quad H(\omega) = \sum_k h_k e^{-ik\omega}.$$

Since $\widehat{\varphi}(0) = 1$, we can apply (8) recursively. This yields, at least formally, the product formula

$$\widehat{\varphi}(\omega) = \prod_{j=1}^{\infty} H(2^{-j}\omega).$$

The convergence of this product is examined in [27, 47]. The representation of $\widehat{\varphi}$ is nice to have in many situations. For example, it can be used to construct $\varphi(x)$ from the h_k . Using (7) and (8), we see that we obtain a partition of unity if

$$H(\pi) = 0 \quad \text{or} \quad \sum_k (-1)^k h_k = 0.$$

Also note that (5) can be written as

$$H(0) = 1.$$

Examples of scaling functions.

(i) A well-known family of scaling functions is the set of cardinal B-splines. The cardinal B-spline of order 1 is the box function $N_1(x) = \chi_{[0,1]}(x)$. For $m > 1$ the cardinal B-spline N_m is defined recursively as a convolution:

$$N_m = N_{m-1} * N_1.$$

These functions satisfy

$$N_m(x) = 2^{m-1} \sum_{k=0}^m \binom{m}{k} N_m(2x-k),$$

and

$$\widehat{N}_m(\omega) = \left(\frac{1 - e^{-i\omega}}{i\omega} \right)^m.$$

(ii) Another classical example is the Shannon sampling function,

$$\varphi(x) = \frac{\sin(\pi x)}{\pi x} \quad \text{with} \quad \widehat{\varphi}(\omega) = \chi_{[-\pi, \pi]}(\omega).$$

We may take

$$H(\omega) = \chi_{[-\pi/2, \pi/2]}(\omega) \quad \text{for} \quad \omega \in [-\pi, \pi],$$

and, consequently,

$$h_{2k} = 1/2 \delta_k \quad \text{and} \quad h_{2k+1} = \frac{(-1)^k}{(2k+1)\pi} \quad \text{for} \quad k \in \mathbb{Z}.$$

Now, for later reference, let us introduce the following 2π -periodic function:

$$F(\omega) = \sum_k |\widehat{\varphi}(\omega + k2\pi)|^2.$$

The fact that φ and its translates form a Riesz basis, corresponds to the fact that there are positive constants A and B such that

$$0 < A \leq F(\omega) \leq B < \infty.$$

Using (8) and rearranging the even and odd terms, we have

$$\begin{aligned} F(2\omega) &= \sum_k |\widehat{\varphi}(2\omega + k2\pi)|^2 \\ &= \sum_k |H(\omega + k\pi)|^2 |\widehat{\varphi}(\omega + k\pi)|^2 \\ &= \sum_k |H(\omega + k2\pi)|^2 |\widehat{\varphi}(\omega + k2\pi)|^2 + |H(\omega + \pi + k2\pi)|^2 |\widehat{\varphi}(\omega + \pi + k2\pi)|^2 \\ (10) \quad &= |H(\omega)|^2 F(\omega) + |H(\omega + \pi)|^2 F(\omega + \pi). \end{aligned}$$

This shows that F is actually π -periodic.

5.2. The wavelet function and the detail spaces W_j . We will use W_j to denote a space complementing V_j in V_{j+1} , i.e. a space that satisfies

$$V_{j+1} = V_j \oplus W_j,$$

where the symbol \oplus stands for direct sum. In other words, each element of V_{j+1} can be written, in a unique way, as the sum of an element of W_j and an element of V_j . We note that the spaces W_j themselves are not necessarily unique; there may be several ways to complement V_j in V_{j+1} .

The space W_j contains the "detail" information needed to go from an approximation at resolution j to an approximation at resolution $j + 1$. Consequently,

$$\bigoplus_j W_j = L^2(\mathbb{R}).$$

A function ψ is a *wavelet* if the collection of functions $\{\psi(x-l) \mid l \in \mathbb{Z}\}$ is a Riesz basis of W_0 . The collection of wavelet functions $\{\psi_{j,l} \mid l, j \in \mathbb{Z}\}$ is then a Riesz basis

of $L^2(\mathbb{R})$. The definition of $\psi_{j,l}$ is similar to the one of $\varphi_{j,l}$ in the previous section. Note that a union of Riesz bases does not necessarily give a Riesz basis for the total span. Even though we did not impose any orthogonality, spaces W_j and $W_{j'}$ are "almost" diagonal for $|j - j'|$ large, and this allows the collection of all $\psi_{j,l}$ to form a Riesz basis for L^2 . Since the wavelet ψ is an element of V_1 , a sequence $(g_k) \in \ell^2(\mathbb{Z})$ exists such that

$$(11) \quad \psi(x) = 2 \sum_k g_k \varphi(2x - k).$$

The Fourier transform of the wavelet is given by

$$(12) \quad \widehat{\psi}(\omega) = G(\omega/2) \widehat{\psi}(\omega/2),$$

where G is a 2π -periodic function given by

$$(13) \quad G(\omega) = \sum_k g_k e^{-ik\omega}.$$

Each space V_j and W_j has a complement in $L^2(\mathbb{R})$ denoted by V_j^c and W_j^c , respectively. We have:

$$V_j^c = \bigoplus_{i=j}^{\infty} W_i \quad \text{and} \quad W_j^c = \bigoplus_{i \neq j} W_i.$$

We define \mathcal{P}_j as the projection operator onto V_j and parallel to V_j^c , and \mathcal{Q}_j as the projection operator onto W_j and parallel to W_j^c . A function f can now be written as

$$f(x) = \sum_j \mathcal{Q}_j f(x) = \sum_{j,l} \gamma_{j,l} \psi_{j,l}(x).$$

Recalling the discussion in Section 4, we see that this last equation is in fact an inverse "discrete" wavelet transform. At this moment the exact conditions on the wavelet are still unclear. They will be made more precise in the next sections. There it will also become clear how to find the coefficients $\gamma_{j,l}$. We first turn to the case where the $\psi_{j,l}$ form an orthonormal basis for $L^2(\mathbb{R})$.

6. Orthogonal wavelets. The class of *orthogonal wavelets* is particularly interesting. We start by introducing the concept of an *orthogonal multiresolution analysis*. This is a multiresolution analysis where the wavelet spaces W_j are defined as the *orthogonal complement* of V_j in V_{j+1} . Consequently, the spaces W_j with $j \in \mathbb{Z}$ are all mutually orthogonal, the projections \mathcal{P}_j and \mathcal{Q}_j are orthogonal, and the expansion

$$f(x) = \sum_j \mathcal{Q}_j f(x)$$

is an orthogonal expansion. A sufficient condition for a multiresolution analysis to be orthogonal is

$$W_0 \perp V_0,$$

or

$$\langle \psi, \varphi(\cdot - l) \rangle = 0 \quad l \in \mathbb{Z},$$

since the other conditions simply follow from scaling. Using Poisson's summation formula, we see that this condition is (essentially) equivalent to

$$(14) \quad \forall \omega \in \mathbf{R} : \sum_k \widehat{\psi}(\omega + k2\pi) \overline{\widehat{\varphi}(\omega + k2\pi)} = 0.$$

An *orthogonal scaling function* is a function φ such that the set $\{\varphi(x-l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis, or

$$(15) \quad \langle \varphi, \varphi(\cdot - l) \rangle = \delta_l \quad l \in \mathbf{Z}.$$

With such a φ , the collection of functions $\{\varphi(x-l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis of V_0 and the collection of functions $\{\varphi_{j,l} \mid l \in \mathbf{Z}\}$ is an orthonormal basis of V_j . Using Poisson's formula, (15) is (essentially) equivalent to

$$(16) \quad \forall \omega \in \mathbf{R} : \sum_k |\widehat{\varphi}(\omega + k2\pi)|^2 = F(\omega) = 1.$$

From (10) we now see that,

$$(17) \quad \forall \omega \in \mathbf{R} : |H(\omega)|^2 + |H(\omega + \pi)|^2 = 1,$$

or

$$\sum_k h_k h_{k-2l} = \delta_l/2 \quad \text{for } l \in \mathbf{Z}.$$

The last two equations are equivalent, but they only provide a necessary condition for the orthogonality of the scaling function and its translates. This relationship is investigated in detail in [28, 94].

Now, an *orthogonal wavelet* is a function ψ such that the collection of functions $\{\psi(x-l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis of W_0 . This is the case if

$$\langle \psi, \psi(\cdot - l) \rangle = \delta_l.$$

Again these conditions are (essentially) equivalent to

$$\forall \omega \in \mathbf{R} : \sum_k |\widehat{\psi}(\omega + k2\pi)|^2 = 1,$$

and, using a similar argument as above, a necessary condition is given by

$$\forall \omega \in \mathbf{R} : |G(\omega)|^2 + |G(\omega + \pi)|^2 = 1.$$

Since the spaces W_j are mutually orthogonal, the collection of functions $\{\psi_{j,l} \mid j, l \in \mathbf{Z}\}$ is an orthonormal basis of $L^2(\mathbf{R})$.

The projection operators \mathcal{P}_j and \mathcal{Q}_j can now be written as

$$\mathcal{P}_j f(x) = \sum_l \langle f, \varphi_{j,l} \rangle \varphi_{j,l}(x) \quad \text{and} \quad \mathcal{Q}_j f(x) = \sum_l \langle f, \psi_{j,l} \rangle \psi_{j,l}(x).$$

They yield the best L^2 approximations of the function f in V_j and W_j , respectively. For a function $f \in L^2(\mathbf{R})$ we have the orthogonal expansion

$$f(x) = \sum_{j,l} \gamma_{j,l} \psi_{j,l}(x) \quad \text{with} \quad \gamma_{j,l} = \langle f, \psi_{j,l} \rangle.$$

Again, this can be viewed as a discrete version of the continuous wavelet transform. Examples of orthogonal wavelets will be given in Section 10.

Using (16) we can write the condition (14) as

$$(18) \quad \forall \omega \in \mathbf{R} : G(\omega) \overline{H(\omega)} + G(\omega + \pi) \overline{H(\omega + \pi)} = 0.$$

From this last equation it follows that the function $G(\omega)$ needs to be of the form

$$G(\omega) = A(\omega) \overline{H(\omega + \pi)},$$

where A is a 2π -periodic function such that

$$A(\omega + \pi) = -A(\omega).$$

The orthogonality of the wavelet immediately follows from the orthogonality of the scaling function if

$$|A(\omega)| = 1.$$

As we will see later on, it is important for the scaling function and wavelet to have compact support. The compact support of the wavelet and scaling function is equivalent with the fact that H and G are trigonometric polynomials (i.e. the sums in (9) and (13) are finite). In the above case, we see that if the scaling function is compactly supported, so is the wavelet, provided that A is a trigonometric polynomial. The only trigonometric polynomials that satisfy the conditions for A are monomials of the form,

$$C e^{-(2k+1)\omega} \text{ with } |C| = 1 \text{ and } k \in \mathbf{Z}.$$

Up to the constant C and an integer translation, the different A all give rise to the same wavelet. Any other choice for A will lead to a wavelet without compact support. If the coefficients h_k are real, so are the g_k if $C = \pm 1$. The standard choice is $A(\omega) = -e^{-i\omega}$. This means that we derive an orthogonal wavelet from an orthogonal scaling function by choosing

$$(19) \quad g_k = (-1)^k \overline{h_{1-k}}.$$

This still leaves us with the problem of constructing a compactly supported scaling function. We will comment on this in Section 8.

In [95] an orthogonalization procedure to find orthonormal wavelets is proposed. It states that if a function φ and its integer translates form a Riesz basis of V_0 , then an orthonormal basis of V_0 is given by φ_{orth} and its integer translates with

$$(20) \quad \varphi_{orth}(\omega) = \frac{\widehat{\varphi}(\omega)}{\sqrt{F(\omega)}}.$$

The fact that we started from a Riesz basis guarantees that $F(\omega)$ is strictly positive. We see that φ indeed satisfies the orthogonality condition (16). Note that if φ is compactly supported, φ_{orth} will, in general, not be compactly supported.

7. Biorthogonal wavelets. The orthogonality property puts a strong limitation on the construction of wavelets. For example, it is known that the Haar wavelet is the only real-valued wavelet that is compactly supported, symmetric and orthogonal [47]. The generalization to *biorthogonal wavelets* has been considered to gain more flexibility. Here, a dual scaling function $\tilde{\varphi}$ and a dual wavelet $\tilde{\psi}$ exist that generate a dual multiresolution analysis with subspaces \tilde{V}_j and \tilde{W}_j , such that

$$(21) \quad \tilde{V}_j \perp W_j \quad \text{and} \quad V_j \perp \tilde{W}_j,$$

and, consequently,

$$\tilde{W}_j \perp W_{j'} \quad \text{for} \quad j \neq j'.$$

The dual multiresolution analysis is not necessarily the same as the one generated by the original basis functions. An equivalent condition to (21) is

$$\langle \tilde{\varphi}, \varphi(\cdot - l) \rangle = \langle \tilde{\psi}, \psi(\cdot - l) \rangle = 0.$$

Moreover, the dual functions also have to satisfy

$$\langle \tilde{\varphi}, \varphi(\cdot - l) \rangle = \delta_l \quad \text{and} \quad \langle \tilde{\psi}, \psi(\cdot - l) \rangle = \delta_l.$$

By using a scaling argument, we have the seemingly more general properties that

$$(22) \quad \langle \tilde{\varphi}_{j,l}, \varphi_{j',l'} \rangle = \delta_{l-l'} \quad l, l', j, j' \in \mathbf{Z}$$

and

$$(23) \quad \langle \tilde{\psi}_{j,l}, \psi_{j',l'} \rangle = \delta_{j-j'} \delta_{l-l'} \quad l, l', j, j' \in \mathbf{Z}.$$

Here the definitions of $\tilde{\varphi}_{j,l}$ and $\tilde{\psi}_{j,l}$ are similar to the ones for $\varphi_{j,l}$ and $\psi_{j,l}$. Note that the role of the basis (i.e. the φ and ψ) and the dual basis can be interchanged. Using the same Fourier techniques as in the previous section, the biorthogonality conditions are (essentially) equivalent with

$$(24) \quad \forall \omega \in \mathbf{R} : \begin{cases} \sum_k \tilde{\varphi}(\omega + k2\pi) \overline{\varphi(\omega + k2\pi)} = 1 \\ \sum_k \tilde{\psi}(\omega + k2\pi) \overline{\psi(\omega + k2\pi)} = 1 \\ \sum_k \tilde{\psi}(\omega + k2\pi) \overline{\varphi(\omega + k2\pi)} = 0 \\ \sum_k \tilde{\varphi}(\omega + k2\pi) \overline{\psi(\omega + k2\pi)} = 0. \end{cases}$$

Since they define a multiresolution analysis, the dual functions must satisfy

$$(25) \quad \tilde{\varphi}(x) = 2 \sum_k \tilde{h}_k \tilde{\varphi}(2x - k) \quad \text{and} \quad \tilde{\psi}(x) = 2 \sum_k \tilde{g}_k \tilde{\varphi}(2x - k).$$

If we define the functions \tilde{H} and \tilde{G} in the same fashion as we did for H and G , then necessary conditions are again given by

$$(26) \quad \forall \omega \in \mathbf{R} : \begin{cases} \tilde{H}(\omega) \overline{\tilde{H}(\omega)} + \tilde{H}(\omega + \pi) \overline{\tilde{H}(\omega + \pi)} = 1 \\ \tilde{G}(\omega) \overline{\tilde{G}(\omega)} + \tilde{G}(\omega + \pi) \overline{\tilde{G}(\omega + \pi)} = 1 \\ \tilde{G}(\omega) \overline{\tilde{H}(\omega)} + \tilde{G}(\omega + \pi) \overline{\tilde{H}(\omega + \pi)} = 0 \\ \tilde{H}(\omega) \overline{\tilde{G}(\omega)} + \tilde{H}(\omega + \pi) \overline{\tilde{G}(\omega + \pi)} = 0, \end{cases}$$

or

$$\forall \omega \in \mathbb{R} : \begin{bmatrix} \tilde{H}(\omega) & \tilde{H}(\omega + \pi) \\ \tilde{G}(\omega) & \tilde{G}(\omega + \pi) \end{bmatrix} \begin{bmatrix} H(\omega) & G(\omega) \\ H(\omega + \pi) & G(\omega + \pi) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Hence, if we let

$$M(\omega) = \begin{bmatrix} H(\omega) & H(\omega + \pi) \\ G(\omega) & G(\omega + \pi) \end{bmatrix},$$

and similarly for \tilde{M} , then

$$\tilde{M}(\omega) \overline{M^t(\omega)} = 1.$$

By interchanging the matrices on the left-hand side, we get

$$(27) \quad \forall \omega \in \mathbb{R} : \begin{cases} \overline{H(\omega)} \tilde{H}(\omega) + \overline{G(\omega)} \tilde{G}(\omega) = 1 \\ \overline{H(\omega)} \tilde{H}(\omega + \pi) + \overline{G(\omega)} \tilde{G}(\omega + \pi) = 0. \end{cases}$$

Note that the orthogonal case corresponds to M being a unitary matrix. Cramer's rule now states that

$$(28) \quad \tilde{H}(\omega) = \frac{\overline{G(\omega + \pi)}}{\Delta(\omega)}$$

and

$$(29) \quad \tilde{G}(\omega) = -\frac{\overline{H(\omega + \pi)}}{\Delta(\omega)},$$

where

$$\Delta(\omega) = \det M(\omega).$$

The fact that the wavelets form a basis for the complementary spaces ensures that Δ does not vanish.

The projection operators take the form

$$\mathcal{P}_j f(x) = \sum_l \langle f, \tilde{\varphi}_{j,l} \rangle \varphi_{j,l}(x) \quad \text{and} \quad \mathcal{Q}_j f(x) = \sum_l \langle f, \tilde{\psi}_{j,l} \rangle \psi_{j,l}(x),$$

and

$$f = \sum_{j,l} \langle f, \tilde{\psi}_{j,l} \rangle \psi_{j,l}.$$

Note that this can be viewed as a "discrete" wavelet transform and that the conditions on ψ are less restrictive than in the orthogonal case. From the equations (22), (23), and (25) we see that

$$\tilde{h}_{k-2l} = \langle \tilde{\varphi}(x-l), \varphi(2x-k) \rangle \quad \text{and} \quad \tilde{g}_{k-2l} = \langle \tilde{\psi}(x-l), \varphi(2x-k) \rangle.$$

In particular, by writing $\varphi(2x-k) \in V_1$ in the bases of V_0 and W_0 , we obtain that

$$(30) \quad \varphi(2x-k) = \sum_l \tilde{h}_{k-2l} \varphi(x-l) + \sum_l \tilde{g}_{k-2l} \psi(x-l).$$

Even if the scaling function and the wavelet are not orthogonal, the multiresolution analysis may still be orthogonal. Let us study this in a little more detail

A biorthogonal scaling function and wavelet are *semiorthogonal* if they generate an orthogonal multiresolution analysis [1, 2, 20]. The name *pre-wavelet* is also used for such a wavelet. Since the W_j subspaces are mutually orthogonal we have that

$$W_j \perp \widetilde{W}_{j'} \text{ and } W_j \perp W_{j'} \text{ for } j \neq j'.$$

Consequently, $W_j = \widetilde{W}_j$, which implies that $V_j = \widetilde{V}_j$. Thus, the primary and dual functions generate the same (orthogonal) multiresolution analysis. A dual scaling function can now be found by letting

$$\widehat{\varphi}(\omega) = \frac{\widetilde{\varphi}(\omega)}{F(\omega)}.$$

We see that the first equation of (24) is satisfied, and, since F is a bounded, 2π -periodic function that does not vanish, the translates of φ and $\widehat{\varphi}$ generate the same space. This corresponds to:

$$\widetilde{H}(\omega) = \frac{H(\omega) F(\omega)}{F(2\omega)}.$$

In order to have an orthogonal multiresolution analysis, (18) must also be satisfied. As before, this means that we need to pick G so that

$$G(\omega) = A(\omega) \overline{H(\omega + \pi)},$$

where A is a 2π -periodic function with

$$A(\omega + \pi) = -A(\omega).$$

If A is a trigonometric polynomial, then the scaling function is compactly supported. By looking at the last equation of (26) it is clear that a simple choice is

$$A(\omega) = -e^{-i\omega} F(\omega + \pi),$$

so that

$$\Delta(\omega) = e^{-i\omega} F(2\omega),$$

and, consequently,

$$\widetilde{G}(\omega) = -e^{-i\omega} \frac{\overline{H(\omega + \pi)}}{F(2\omega)}.$$

If φ is a compactly supported function, this construction guarantees that ψ is compactly supported too, since H and F , and hence also G , are trigonometric polynomials. However, the dual functions are, in general, not compactly supported.

8. Wavelets and polynomials. The moments of the scaling function and wavelet are defined by:

$$\mathcal{M}_p = \int_{-\infty}^{+\infty} x^p \varphi(x) dx \text{ and } \mathcal{N}_p = \int_{-\infty}^{+\infty} x^p \psi(x) dx \text{ with } p \in \mathbb{N},$$

and similarly for the dual functions. The scaling functions are normalized with $\mathcal{M}_0 = \widetilde{\mathcal{M}}_0 = 1$.

Recall that we want the scaling function to satisfy a "partition of unity" property and, furthermore, that this corresponds to $H(\pi) = 0$. From (29) we see that this implies that $\widetilde{G}(0) = 0$ and, hence, that $\widetilde{N}_0 = 0$. So the dual wavelet needs to have a vanishing integral. This is reminiscent of the case of the continuous wavelet transform where we needed the wavelet to have a vanishing integral.

As we pointed out before, the fact that the wavelet has a vanishing integral allows us to give a precise characterization of the functions with a certain smoothness (when the order of smoothness α is less than 1), in terms of the decay of the continuous wavelet transform. The analogous fact is true here: the wavelet coefficients are given by inner products with the dual wavelets and the fact that these have a vanishing integral allows us to characterize exactly which functions will be of a certain smoothness by looking at the decay of the coefficients.

As in the case of the continuous wavelet transform, to obtain similar characterizations of classes of functions of smoothness $\alpha > 1$, the dual wavelet needs to have more vanishing moments. This is in fact closely related to the property that the scaling function and its translates can be used to represent polynomials. We make this statement more precise.

Let N denote the number of vanishing moments of the dual wavelet,

$$\widetilde{N}_p = 0 \text{ for } 0 \leq p < N \text{ and } \widetilde{N}_N \neq 0.$$

This is the same as saying that $\widehat{\psi}(\omega)$ has a root of multiplicity N at $\omega = 0$. Since $\widehat{\varphi}(0) \neq 0$, it is also equivalent to the fact that $\widetilde{G}(\omega)$ has a root of multiplicity N at $\omega = 0$. Thus, the sequence $\{\widetilde{g}_k\}$ also has N vanishing discrete moments,

$$\sum_k \widetilde{g}_k k^p = 0, \text{ for } 0 \leq p < N.$$

From (29), we see that this is equivalent to $H(\omega)$ having a root of multiplicity N at $\omega = \pi$, which, by using (8), implies that

$$(31) \quad i^p \widehat{\varphi}^{(p)}(2k\pi) = \delta_k \mathcal{M}_p \text{ for } 0 \leq p < N.$$

By Poisson's summation formula, it follows that

$$(32) \quad \sum_l (x-l)^p \varphi(x-l) = \mathcal{M}_p \text{ for } 0 \leq p < N.$$

By rearranging the last expression, we see that any polynomial with degree smaller than N can be written as a linear combination of the functions $\varphi(x-l)$ with $l \in \mathbb{Z}$.

At this point we digress a little and make two small remarks.

1. The fact that $H(\omega)$ has a root of multiplicity N at $\omega = \pi$ means that we can factor $H(\omega)$ as

$$H(\omega) = \left(\frac{1 + e^{-i\omega}}{2} \right)^N K(\omega),$$

with $K(0) = 1$ and $K(\pi) \neq 0$. This factorization together with the (bi)orthogonality conditions and the fact that K is a trigonometric polynomial is used as a starting point for the construction of compactly supported wavelets [31, 47].

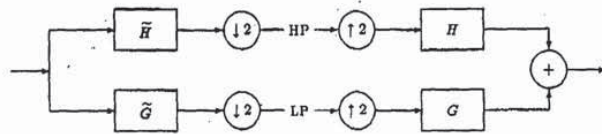


FIG. 1. The subband filtering scheme.

2. When writing a polynomial as a linear combination of the $\varphi(x - l)$, the coefficients in the linear combination themselves are polynomials of the same degree in l . More precisely, if A is a polynomial of degree $p \leq N - 1$, then a polynomial B , of the same degree, exists such that

$$(33) \quad A(x) = \sum_l B(l) \varphi(x - l).$$

The fact that B is indeed a polynomial can easily be seen from

$$B(l) = \int A(x) \tilde{\varphi}(x - l) dx = \int A(x + l) \tilde{\varphi}(x) dx.$$

Furthermore,

$$A(x) = \sum_l B(x - l) \varphi(l),$$

since the polynomials on the left and right-hand sides match at each integer.

With the extra vanishing moment conditions on the dual wavelet, we can characterize smoothness up to order $\alpha < N$. Another consequence is that the convergence rate of the wavelet approximation for smooth functions now immediately follows: if $f \in C^N$, then

$$(34) \quad \|\mathcal{P}_j f(x) - f(x)\| = \mathcal{O}(h^N) \quad \text{with } h = 2^{-j}.$$

The conditions (31) are referred to as the Strang-Fix conditions, and these were established long before the development of wavelet theory [69, 122, 124].

An asymptotic error expansion in powers of h , which can be used in numerical extrapolation, is derived in [127, 128]. For results on the pointwise convergence properties of wavelet series, see [90].

The exponent N in the factorization of H also plays a role in the regularity of φ . The Hölder regularity is $N - 1$ at most, but in many cases it is lower due to the influence of K . The regularity of solutions of refinement equations is studied in detail in [42, 41, 52, 53, 66, 114, 135, 136].

Note that we never required the dual scaling function to satisfy a partition of unity property, nor the wavelet to have a vanishing moment. In fact, it is possible to have a wavelet with a non-vanishing integral. In that case the regularity of the dual functions is very low. It may even be that they are distributions instead of functions, but this is not necessarily a problem in applications.

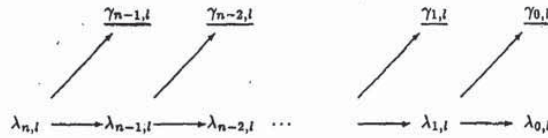


FIG. 2. The decomposition scheme.

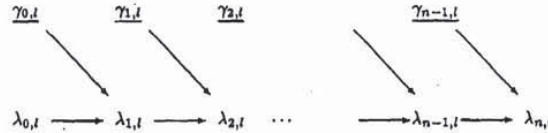


FIG. 3. The reconstruction scheme.

9. **The fast wavelet transform.** Since V_j is equal to $V_{j-1} \oplus W_{j-1}$, a function $v_j \in V_j$ can be written uniquely as the sum of a function $v_{j-1} \in V_{j-1}$ and a function $w_{j-1} \in W_{j-1}$:

$$\begin{aligned} v_j(x) &= \sum_k \lambda_{j,k} \varphi_{j,k}(x) = v_{j-1}(x) + w_{j-1}(x) \\ &= \sum_l \lambda_{j-1,l} \varphi_{j-1,l}(x) + \sum_l \gamma_{j-1,l} \psi_{j-1,l}(x). \end{aligned}$$

In other words, we have two representations of the function v_j , one as an element in V_j and associated with the sequence $\{\lambda_{j,k}\}$, and another as a sum of elements in V_{j-1} and W_{j-1} and associated with the sequences $\{\lambda_{j-1,k}\}$ and $\{\gamma_{j-1,k}\}$. The following relations show how to pass between these representations. By (25),

$$\begin{aligned} \lambda_{j-1,t} &= \langle v_j, \tilde{\varphi}_{j-1,t} \rangle = \sqrt{2} \langle v_j, \sum_k \tilde{h}_{k-2l} \tilde{\varphi}_{j,k} \rangle \\ (35) \quad &= \sqrt{2} \sum_k \tilde{h}_{k-2l} \lambda_{j,k}, \end{aligned}$$

and, similarly,

$$(36) \quad \gamma_{j-1,t} = \sqrt{2} \sum_k \tilde{g}_{k-2l} \lambda_{j,k}.$$

The opposite direction, from the $\lambda_{j-1,t}$ and the $\gamma_{j-1,t}$ to the $\lambda_{j,k}$, is equally easy. Using (30) we have

$$(37) \quad \lambda_{j,k} = \sqrt{2} \sum_l h_{k-2l} \lambda_{j-1,l} + \sqrt{2} \sum_l g_{k-2l} \gamma_{j-1,l}.$$